Forces and stress in second order Møller-Plesset perturbation theory for condensed phase systems within the resolution-of-identity Gaussian and plane waves approach

Del Ben, Mauro ; Hutter, Jürg ; VandeVondele, Joost

Abstract: The forces acting on the atoms as well as the stress tensor are crucial ingredients for calculating the structural and dynamical properties of systems in the condensed phase. Here, these derivatives of the total energy are evaluated for the second-order Møller-Plesset perturbation energy (MP2) in the framework of the resolution of identity Gaussian and plane waves method, in a way that is fully consistent with how the total energy is computed. This consistency is non-trivial, given the different ways employed to compute Coulomb, exchange, and canonical four center integrals, and allows, for example, for energy conserving dynamics in various ensembles. Based on this formalism, a massively parallel algorithm has been developed for finite and extended system. The designed parallel algorithm displays, with respect to the system size, cubic, quartic, and quintic requirements, respectively, for the memory, communication, and computation. All these requirements are reduced with an increasing number of processes, and the measured performance shows excellent parallel scalability and efficiency up to thousands of nodes. Additionally, the computationally more demanding quintic scaling steps can be accelerated by employing graphics processing units (GPU's) showing, for large systems, a gain of almost a factor two compared to the standard central processing unit-only case. In this way, the evaluation of the derivatives of the RI-MP2 energy can be performed within a few minutes for systems containing hundreds of atoms and thousands of basis functions. With good time to solution, the implementation thus opens the possibility to perform molecular dynamics (MD) simulations in various ensembles (microcanonical ensemble and isobaric-isothermal ensemble) at the MP2 level of theory. Geometry optimization, full cell relaxation, and energy conserving MD simulations have been performed for a variety of molecular crystals including NH3, CO2, formic acid, and benzene.

DOI: https://doi.org/10.1063/1.4919238

Forces and Stress in Second Order Møller-Plesset Perturbation Theory for Condensed Phase Systems within the Resolution-of-Identity Gaussian and Plane Waves Approach

Mauro Del Ben[a)] and Jürg Hutter[b)]

Department of Chemistry, University of Zürich, Winterthurerstrasse 190, CH-8057, Zürich, Switzerland

Joost VandeVondele[c)]

Department of Materials, ETH Zürich, Wolfgang-Pauli-Strasse 27, CH-8093 Zürich, Switzerland

The forces acting on the atoms as well as the stress tensor are crucial ingredients for calculating the structural and dynamical properties of systems in the condensed phase. Here, these derivatives of the total energy are evaluated for the second-order Møller-Plesset perturbation energy (MP2) in the framework of the Resolution of Identity Gaussian and Plane Waves (RI-GPW) method, in a way that is fully consistent with how the total energy is computed. This consistency is non-trivial, given the different ways employed to compute Coulomb, exchange and canonical four center integrals, and allows, for example, for energy conserving dynamics in various ensembles. Based on this formalism, a massively parallel algorithm has been developed for finite and extended system. The designed parallel algorithm displays, with respect of the system size, cubic, quartic and quintic requirements respectively for the memory, communication and computation. All these requirements are reduced with an increasing number of processes and the measured performance shows excellent parallel scalability and efficiency up to thousands of nodes. Additionally, the computationally more demanding quintic scaling steps can be accelerated by employing graphics processing units (GPU) showing, for large systems, a gain of almost a factor two compared to the standard CPU-only case. In this way, the evaluation of the derivatives of the RI-MP2 energy can be performed within a few minutes for systems containing hundreds of atoms and thousands of basis functions. With good time to solution, the implementation thus opens the possibility to perform molecular dynamics simulations in various ensembles (NVE, NpT) at the MP2 level of theory. Geometry optimization, full cell relaxation, and energy conserving MD simulations have been performed for a variety of molecular crystals including NH_3, CO_2, Formic Acid and Benzene.

Keywords: Gradients, Forces, Stress, Gaussian and Plane Waves, Resolution-of-Identity, Møller-Plesset Perturbation Theory, MP2, Condensed Phase, Molecular Crystals

[a)]Electronic mail: mauro.delben@chem.uzh.ch

[b)]Electronic mail: hutter@chem.uzh.ch

[c)]Electronic mail: Joost.VandeVondele@mat.ethz.ch

I. INTRODUCTION

The energy evaluated with the second-order Møller-Plesset (MP2) perturbation theory represents an effective way to improve the Hartree-Fock (HF) ground state by including electron correlation effects.[1,2] In this respect, MP2, also referred as second-order many body perturbation theory (MBPT(2)), offers many appealing features, such as size consistency and the capability to correctly account for dispersion interactions.[3] In addition to that, MP2 is an *ab-initio* method that can accurately describe hydrogen-bond, covalent and ionic interactions from first principles. Moreover, among the correlated electronic structure methods, MP2 is probably that one displaying the simplest and most compact form. For these reasons MP2 is often used as a reference for testing and benchmarking new approximate methods and MP2-like correlation has also been included in Density Functional Theory (DFT) with the introduction of Double-Hybrid Density Functionals.[4,5] However, despite the advantages of MP2, the inherent computational cost has limited its use. This is due to the quintic growth of the computational effort with respect to systems size. Additionally, the basis set size must be large in order to represent the electron coalescence cusp and converge the MP2 energy.[6,7] During the past decades, several groups have contributed to improving this situation, and to extending the applicability of MP2 in various ways.[8]

Several approaches have been proposed in order to reduce the formal $O(N^5)$ scaling and they can be classified as Laplace-Transformed MP2,[9–16] Local MP2 (LMP2),[17–26] and Stochastic[27–30] methods, while explicitly correlated schemes can be used for accelerating the convergence of the MP2 energy with respect to basis set size (F12-MP2).[31–33] Furthermore, the Resolution of Identity (RI)[34–42] approximation, sometimes referred as Density Fitting (DF), has shown to greatly speed up the evaluation of the MP2 energy giving almost a order of magnitude reduction of the computational cost without significant loss of accuracy.[43–45] In addition, parallel computing has become of prime importance in quantum chemistry as a tool for reducing the time to solution for these calculations. In this respect several parallel algorithms have been proposed[46–59] showing an efficiency growing at the same rate as the increase of the computational power. Recently, for the related direct random phase approximation method, we have demonstrated the feasibility of computing the correlation energy of more than a thousand atoms, with good basis sets, within hours.[60]

Thanks to all these improvements the applicability of MP2 theory has been increased

continuously over time and recently an RI-MP2 Monte Carlo (MC) simulation under ambient conditions of bulk liquid water has been reported[61,62], demonstrating the feasibility of computing tens of thousands of configurations of 64 molecules within a reasonable time. The advantage of the MC scheme is that only the total energy is required in order to calculate ensemble averages. On the other hand, an efficient MC sampling needs a sufficient knowledge of the system under study necessary to define "smart" trial move. This makes the use of MC methods less straightforward than Molecular Dynamics (MD), for which the ensemble averages are obtained by integrating the classical equations of motion. In this case the forces acting on atoms have to be computed, obtained as partial derivatives of the total energy with respect to the atomic positions. Furthermore, MC does not give access to truly dynamical properties, i.e. derived from time correlation functions, such as for example, diffusion constants and vibrational spectroscopy. To obtain those, accurate energy conserving (NVE) simulations have to be performed, requiring consistent forces.

The evaluation of the derivatives at the MP2 level is more intricate compared to their computation at the HF or DFT level. This is because, contrary to the HF or DFT cases, the correlation energy obtained from perturbation theory is not stationary with respect to the molecular orbital expansion coefficients, implying that first order orbital response has to be computed. The theory and equations for calculating the energy derivatives at the MP2[63–66] and RI-MP2[67] level have been derived and reported by many authors, together with many serial[68–70] and parallel[71–74] implementations.

Here the equations for evaluating the derivatives of the second-order Møller-Plesset perturbation energy in the framework of the Resolution of Identity Gaussian and Plane Waves (RI-GPW) are presented. The derivatives are evaluated consistently to the way the RI-MP2 energy is computed,[59] and are of general validity for both finite and extended systems. The central idea in the RI approximation is the introductions of an atom-center Gaussian auxiliary basis used for fitting pairwise products of atomic orbital basis functions. In addition to the representation in term of Gaussian functions, the RI-fitting densities within the RI-GPW method are expressed also employing an auxiliary basis of Plane Waves (PW). This choice allows for rapid conversion between direct and reciprocal space representation of the density by employing fast Fourier transformations (FFT). In this way the treatment of the Coulomb interactions is efficiently accomplished by integration of the electrostatic potential associated to each RI-fitting density over the pairs of primary basis functions,

where the electrostatic potential is obtained in a plane wave basis set after the solution of the Poisson equation in Fourier space. As a drawback, the GPW method requires smooth densities, implying that pseudopotentials have to be employed. All-electron calculations are possible within the Gaussian and Augmented Plane Wave (GAPW) scheme,[75,76] however the actual implementation is currently limited to the GPW method only.

An implementation of the analytical energy gradients at the MP2 level for extended systems has been reported by Hirata and coworkers.[77] In this case the formulation is based on the crystal orbital theory implying that two-electron integrals are obtained by k-point sampling in the first Brillouin zone. Moreover the reported applications are limited to polymers (periodic 1D) with small basis sets. The difference compared to the method presented here relies in the way the two-electron integrals are computed. In our current implementation, the sampling of the first Brillouin zone is restricted to the Gamma point only and thus the approach converges to the same value as obtained from full k-point sampling if a sufficiently large supercell is chosen. Since our aim is to enable the study of large and disordered systems, our approach is suitable for applications.

For the presented scheme a massively parallel algorithm has been designed and implemented in CP2K.[78] The parallel algorithm displays, with respect of the system size, cubic, quartic and quintic effort respectively for the memory, communication and computation. All these requirements scale increasing the number of processes and the measured performance displays excellent parallel scalability and efficiency up to thousands of nodes. Moreover, in the actual implementation the computationally more demanding part, that is the quintic scaling steps, can be accelerated by employing graphics processing units (GPU). Compared to the standard CPU only case, this leads, in general, to a speedup of a factor grater than 4 for the $O(N^5)$ parts of the algorithm, resulting, for the largest cases, in an almost factor 2 reduction in the overall time for the calculation.

Several benchmark calculations are reported with a particular focus on molecular crystals including NH_3, CO_2, Formic Acid and Benzene. In general it has been observed that the effort for the calculation of the derivatives at the RI-MP2 level is between 4 and 5 times more expensive than computing only the energy.

II. THEORY

In this section the basic equations necessary for implementing the first derivatives of the RI-MP2 energy are briefly presented referring to the original works for more details.[66,67,70,74] More information is reported in cases for which the general theory is combined with the GPW approach. The following index notation has been adopted: $i, j, k, \ldots$ refer to canonical occupied molecular orbitals (MOs), $a, b, c, \ldots$ to canonical virtual MOs, $p, q, r, \ldots$ to general canonical MOs, $\mu, \nu, \lambda, \ldots$ to primary atomic orbital basis set functions (AO), $P, Q, R, \ldots$ to auxiliary AO basis set functions (AUX). The one electron MO, primary AO and auxiliary AO functions are symbolized respectively with ψ, ϕ and χ. The number of occupied and virtual orbitals is denoted by o and v, while the total number of primary and auxiliary basis functions as n and N_a. In order to express, in general, the system size, the symbol N is used. Given a perturbation parameter x, *e.g.* a nuclear displacement, the superscript ^{x} represents the derivative with respect to x, while $^{(x)}$ denotes the skeleton derivative, that is derivatives of the AO integrals only (*i.e.* without considering the derivatives of the expansion coefficients of the MOs).

The MP2 Energy within the RI-GPW Method

In Second Order Møller-Plesset perturbation theory, the correlation energy $E^{(2)}$ for a closed shell restricted Hartree-Fock reference wave function is given by:

$$E^{(2)} = -\sum_{i \leq j}^{o} (2 - \delta_{ij}) \sum_{ab}^{v} \frac{(ia|jb)[2(ia|jb) - (ib|ja)]}{\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j} \tag{1}$$

where ϵ_p are orbital energies, δ_{ij} is the Kronecker delta and $(ia|jb)$ is a two-electron repulsion integral (ERI) over MO in Mulliken notation

$$(ia|jb) = \int\int \psi_i(\vec{r}_1)\psi_a(\vec{r}_1)\frac{1}{|\vec{r}_1 - \vec{r}_2|}\psi_j(\vec{r}_2)\psi_b(\vec{r}_2)d\vec{r}_1 d\vec{r}_2. \tag{2}$$

In a standard canonical MP2 energy algorithm the computation of the $(ia|jb)$ integrals is

performed via four consecutive integral transformations of the ERIs over AO $(\mu\nu|\lambda\sigma)$:

$$(ia|jb) = \sum_{\mu} C_{\mu i} \sum_{\nu} C_{\nu a} \sum_{\lambda} C_{\lambda j} \sum_{\sigma} C_{\sigma b} (\mu\nu|\lambda\sigma), \tag{3}$$

where the $C_{\kappa p}$ represent elements of the MO coefficient matrix. Each of the four quarter transformations has a formal computational effort that grows as $O(N^5)$ that eventually reflects into the asymptotic scaling associated to the evaluation of MP2 energy.[79]

The resolution of identity approximation[80,81] is an effective technique that allows to accelerate the evaluation of the $(ia|jb)$ ERIs. It consists in the introduction of an auxiliary Gaussian basis set $\{\chi_P\}$ used to factorize the $(ia|jb)$ integrals according to:

$$(ia|jb)_{\mathrm{RI}} = \sum_{P} B_{ia}^{P} B_{jb}^{P}, \tag{4}$$

where $\mathbf{B}$ is a matrix with ov rows and N_a columns given by:

$$B_{ia}^{P} = \sum_{Q} (ia|Q) V_{QP}^{-1/2} \tag{5}$$

and $V_{QP}^{-1/2}$ are the matrix elements of the inverse square root of the Coulomb metric[82] $V_{QP} = (Q|P)$

$$(Q|P) = \int\int \chi_Q(\vec{r}_1) \frac{1}{|\vec{r}_1 - \vec{r}_2|} \chi_P(\vec{r}_2) d\vec{r}_1 d\vec{r}_2. \tag{6}$$

Since the three center integrals $(ia|Q)$ are computed starting from integrals over AO

$$(\mu\nu|Q) = \int\int \phi_\mu(\vec{r}_1)\phi_\nu(\vec{r}_1) \frac{1}{|\vec{r}_1 - \vec{r}_2|} \chi_Q(\vec{r}_2) d\vec{r}_1 d\vec{r}_2, \tag{7}$$

the final expression for the B_{ia}^{P} elements reads:

$$B_{ia}^{P} = \sum_{\nu} C_{\nu a} \sum_{\mu} C_{\mu i} \sum_{Q} (\mu\nu|Q) V_{QP}^{-1/2}. \tag{8}$$

The RI approximation to the ERIs over MO has many practical advantages:

- The $(ia|jb)_{\mathrm{RI}}$ ERIs can be evaluated without significant loss of accuracy even employing an auxiliary basis that is only $2-4$ times larger than the primary basis.[43,45,67]

- The effort for the integral computation is strongly reduced since 4 index integrals over AO $(\mu\nu|\lambda\sigma)$ are never generated and only three $(\mu\nu|Q)$ and two $(Q|P)$ center ERIs are required. This means that, the integral computation requires formally $O(N^3)$ operations while the integral transformations (equation 8) scale asymptotically as $O(N^4)$.

- As shown by equation 4, the whole set of four index ERIs over MO $(ia|jb)_{\mathrm{RI}}$ can be efficiently evaluated from the three index intermediates B_{ia}^{P} by matrix-matrix multiplications.

- Since for the generation of the $(ia|jb)_{\mathrm{RI}}$ only the matrix $\mathbf{B}$ has to be stored, the required memory grows as $O(N^3)$.

The application of the RI approximation to the MP2 energy calculation is straightforward.[35] It consists simply in the replacement of the $(ia|jb)$ integrals in equation 1 with the approximated $(ia|jb)_{\mathrm{RI}}$ given in equation 4. The computation of the $(ia|jb)_{\mathrm{RI}}$ requires $O(o^2v^2N_a)$ operations implying that the RI-MP2 method is also scaling $O(N^5)$. Thus the advantage of RI-MP2, compared to a standard MP2 implementation, relies on the reduced required memory and prefactor associated to the computation of the $(ia|jb)$ via equation 4 instead of equation 3.

According to what shown so far, it appears clear that applying the RI approximation to the MO-ERIs requires the computation of three $(\mu\nu|Q)$ and two $(Q|P)$ center ERIs. In particular for condensed phase systems, for which periodic boundary conditions (PBC) are mandatory, these intermediates have to account for the requirement that the simulation cell is infinitely replicated in all directions in space. In order to accomplish this task, the Gaussian and Plane-Waves method[83,84] has been extended for handling integrals that arise in wave-function correlation methods.[58,59]

In general, the GPW method is an efficient approach for treating Coulomb interactions between Gaussian basis elements and any electrostatic density ρ that fulfill the PBC of the considered system:

$$(\mu\nu|\rho) = \int\int \phi_\mu(\vec{r}_1)\phi_\nu(\vec{r}_1)\frac{1}{|\vec{r}_1 - \vec{r}_2|}\rho(\vec{r}_2)d\vec{r}_1 d\vec{r}_2. \tag{9}$$

Here, if ρ is considered as the total electronic density, then the form of the last equation is

essentially identical to the one used to compute matrix elements of the Hartree potential.[83] Thus, the highly efficient implementation of that operation in CP2K[78] can be directly used and we refer to Ref.[84] for a detailed discussion.

In particular for the RI case, focusing on three center integrals, they are computed, equation 8, starting from the integrals over AOs that are subsequently transformed with the two matrices $\mathbf{C}$ and $\mathbf{V^{-1/2}}$. Employing the GPW method, equation 9, the index transformation over the auxiliary basis can be avoided, since it is possible to directly compute half transformed integrals for an associated density ρ^P as

$$\begin{aligned} B^P_{\mu\nu} &= \sum_Q (\mu\nu|Q) V^{-1/2}_{QP} \\ &= \int\int \phi_\mu(\vec{r}_1)\phi_\nu(\vec{r}_1)\frac{1}{\vec{r}_{12}}\left[\sum_Q \chi_Q(\vec{r}_2)V^{-1/2}_{QP}\right] d\vec{r}_1 d\vec{r}_2 \\ &= \int \phi_\mu(\vec{r}_1)\phi_\nu(\vec{r}_1)\left[\int \frac{\rho^P(\vec{r}_2)}{\vec{r}_{12}} d\vec{r}_2\right] d\vec{r}_1 \\ &= \int \phi_\mu(\vec{r}_1)\phi_\nu(\vec{r}_1) v^P(\vec{r}_1) d\vec{r}_1. \end{aligned} \tag{10}$$

The same approach holds for the $(P|Q)$ integrals with the only difference that the potential is calculate from the density associated to a single Gaussian auxiliary basis function.

The key aspect in GPW is that the density ρ^P is expressed on a regular grid, or, in equivalent terms, ρ^P is expanded in an auxiliary basis of plane waves (PW)

$$\rho^P(\vec{R}) \approx \frac{1}{\Omega}\sum_{\vec{G}} \rho^P(\vec{G}) e^{i\vec{G}\cdot\vec{R}} \tag{11}$$

where $\rho^P(\vec{G})$ are the Fourier coefficients of the density, Ω is the volume of the simulation cell and the sum over the reciprocal lattice vectors $\vec{G}$ is determined by the size S of the PW basis. Fast Fourier transforms (FFTs) allow for switching representation between real space ($\rho^P(\vec{R})$) and reciprocal ($\rho^P(\vec{G})$) space with an associated computational effort that grows only as $O(S \log S)$. In this way, the electrostatic potential v^P in equation 10 can be efficiently obtained in a plane waves basis set after solving the Poisson equation in Fourier space

$$v^P(\vec{G}) = \frac{4\pi}{G^2}\rho^P(\vec{G}), \tag{12}$$

while an additional back FFT (FFT^{-1}) will yield the potential in real space.

An extensive description, together with the implementation details, of the RI-GPW method can found in Ref.[59], here only the most important features are highlighted:

- The accuracy of the calculated $B^P_{\mu\nu}$ integrals in equation 10 can be systematically improved by increasing the PW basis set size (resolution of the grid).[58] This is conventionally done by specifying the energy cutoff that limits the kinetic energy of the PWs.

- PW auxiliary basis is a natural choice for periodic systems, but it can equally be used for gas phase or surface calculations.[85–87]

- All-electron calculations are not possible and pseudopotentials have to be employed.[88]

- For each electrostatic potential v^P all matrix elements that are non-zero within a given threshold (ϵ_{grid}) can be obtained in linear scaling time.[84]

- $B^P_{\mu\nu}$ are transform from the AO basis to the MO basis (B^P_{ia}) via two consecutive matrix-matrix multiplications, $\mathbf{B}^P_{\mathrm{MO}} = \mathbf{C}_o^\dagger \mathbf{B}^P_{\mathrm{AO}} \mathbf{C}_v$, with $\mathbf{C}_o$ and $\mathbf{C}_v$ being respectively the occupied and virtual parts of the coefficient matrix. The multiplication by $\mathbf{C}_o$ can exploit the sparsity of $\mathbf{B}^P_{\mathrm{AO}}$, implying an $O(no)$ scaling for each P, while the final multiplication can not exploit sparsity and is asymptotically dominant, scaling as $O(onv)$.

The Analytic Derivatives for RI-MP2

The analytic derivative of the RI-MP2 energy $E^{(2)}_{\mathrm{RI}}$ with respect to a perturbation parameter x, for a closed-shell restricted Hartree-Fock wave function, is given by:[67,70,74]

$$\begin{aligned} E^{(2)x}_{\mathrm{RI}} = \frac{dE^{(2)}_{\mathrm{RI}}}{dx} =& 4 \sum_{Q}^{\mathrm{AUX}} \sum_{\mu\nu}^{\mathrm{AO}} \Gamma^Q_{\mu\nu} (\mu\nu|Q)^x - 2 \sum_{PQ}^{\mathrm{AUX}} \Gamma^{PQ} (P|Q)^x \\ &+ 2 \sum_{pq}^{\mathrm{MO}} \left[P^{(2)}_{pq} F^{(x)}_{pq} - W^{(2)}_{pq} S^{(x)}_{pq} \right]. \end{aligned} \tag{13}$$

In the above expression, for each summation, a common structure can be recognized, that is the contraction of terms involving AO derivatives $(\mu\nu|Q)^x, (P|Q)^x, F^{(x)}_{pq}, S^{(x)}_{pq}$, with elements

of the intermediates $\Gamma^Q_{\mu\nu}, \Gamma^{PQ}, P^{(2)}_{pq}, W^{(2)}_{pq}$. The contribution to the derivatives of $E^{(2)}_{\mathrm{RI}}$ coming from the first two summations, referred as non-separable part, is specific to the RI-MP2 method. It involves the contraction of 3- and 2-center RI integral derivative $(\mu\nu|Q)^x, (P|Q)^x$ with 3- and 2-index quantities namely non-separable correction to the 2-particle density matrix (2-PDM), $\Gamma^Q_{\mu\nu}$ and Γ^{PQ}. These two specific quantities are given by:

$$\Gamma^Q_{\mu\nu} = \sum_{i}^{occ} C_{i\mu} \sum_{a}^{virt} C_{\nu a} \Gamma^Q_{ia} \tag{14}$$

$$\Gamma^Q_{ia} = \sum_{P}^{\mathrm{AUX}} Y^P_{ia} V^{-1/2}_{PQ} \tag{15}$$

$$Y^P_{ia} = \sum_{jb} \frac{2(ia|jb) - (ib|ja)}{\epsilon_i + \epsilon_j - \epsilon_a - \epsilon_b} B^P_{jb} \tag{16}$$

$$\Gamma^{PQ} = \sum_{R}^{\mathrm{AUX}} \sum_{ia} \Gamma^P_{ia} B^R_{ia} V^{-1/2}_{RQ} \tag{17}$$

Once $\Gamma^Q_{\mu\nu}$ and Γ^{PQ} are made available, the non-separable contribution to $E^{(2)^x}_{\mathrm{RI}}$ is obtained by contraction with $(\mu\nu|Q)^x$ and $(P|Q)^x$, which are computed consistently to $(\mu\nu|Q)$ and $(P|Q)$ by employing the same GPW scheme. This leads, for the 3-center case, to:

$$\begin{aligned}
\sum_{Q}^{\mathrm{AUX}} \sum_{\mu\nu}^{\mathrm{AO}} \Gamma^Q_{\mu\nu} (\mu\nu|Q)^x &= \sum_{Q}^{\mathrm{AUX}} \sum_{\mu\nu}^{\mathrm{AO}} \Gamma^Q_{\mu\nu} \left[(\mu^x\nu|Q) + (\mu\nu^x|Q) \right] + \sum_{Q}^{\mathrm{AUX}} \left(\sum_{\mu\nu}^{\mathrm{AO}} \Gamma^Q_{\mu\nu} \phi_\mu \phi_\nu | Q^x \right) \\
&= \sum_{Q}^{\mathrm{AUX}} \sum_{\mu\nu}^{\mathrm{AO}} \Gamma^Q_{\mu\nu} \left[(\mu^x\nu|\rho^Q) + (\mu\nu^x|\rho^Q) \right] + \sum_{Q}^{\mathrm{AUX}} (\rho^{\Gamma^Q}|Q^x) \\
&= \sum_{Q}^{\mathrm{AUX}} \sum_{\mu\nu}^{\mathrm{AO}} \Gamma^Q_{\mu\nu} \int \left[\phi^x_\mu(\vec{r}) \phi_\nu(\vec{r}) + \phi_\mu(\vec{r}) \phi^x_\nu(\vec{r}) \right] v^Q_H(\vec{r}) d\vec{r} \\
&\quad + \sum_{Q}^{\mathrm{AUX}} \int \chi^x_Q(\vec{r}) v^{\Gamma^Q}_H(\vec{r}) d\vec{r}
\end{aligned} \tag{18}$$

where $v^{\Gamma^Q}_H$ is the electrostatic potential related to the $\sum_{\mu\nu}^{\mathrm{AO}} \Gamma^Q_{\mu\nu} \phi_\mu(\vec{r}) \phi_\nu(\vec{r})$ density, while v^Q_H is the counterpart associated to the single auxiliary Gaussian function $\chi_Q(\vec{r})$. For the 2-center

case, exploiting the fact that Γ^{PQ} is symmetric, the analogous approach gives:

$$\sum_{PQ}^{\text{AUX}} \Gamma^{PQ}(P|Q)^x = 2\sum_{PQ}^{\text{AUX}} \Gamma^{PQ} \int \chi_P^x(\vec{r}) v_H^Q(\vec{r}) d\vec{r} = 2\sum_{Q}^{\text{AUX}} \int \chi_Q^x(\vec{r}) v_H^{\Gamma^Q}(\vec{r}) d\vec{r} \tag{19}$$

where $v_H^{\Gamma^Q}$ is the potential obtained from the $\sum_P \Gamma^{PQ}\chi_P(\vec{r})$ density, while v_H^Q is the same as in equation 18. The two formulations given in equation 19 are equivalent and, even if the latter offers more advantages in term of computational efficiency, the former is preferred since v_H^Q can be reused in equation 18.

The last summation in equation 13 consists in the contraction of $P_{pq}^{(2)}$, the MP2 correction to the 1-particle density matrix (1-PDM), and $W_{pq}^{(2)}$, the MP2 correction to the energy-weighted density matrix, with the skeleton derivatives of the Fock and overlap matrix elements

$$F_{pq}^{(x)} = \sum_{\mu\nu}^{\text{AO}} C_{\mu p} \left[h_{\mu\nu}^x + \sum_{\lambda\sigma} P_{\lambda\sigma}^{\text{HF}} (\mu\nu|\lambda\sigma)^x - \frac{1}{2}\sum_{\lambda\sigma} P_{\lambda\sigma}^{\text{HF}} (\mu\lambda|\nu\sigma)^x \right] C_{\nu q} \tag{20}$$

$$S_{pq}^{(x)} = \sum_{\mu\nu}^{\text{AO}} C_{\mu p} S_{\mu\nu}^x C_{\nu q}. \tag{21}$$

In equation 20, $h_{\mu\nu}^x$ and $(\mu\nu|\lambda\sigma)^x$ are respectively the derivatives of the one-electron Hamiltonian integrals and the 4-index ERIs over AO, while $P_{\mu\nu}^{\text{HF}} = 2\sum_i^{occ} C_{\mu i} C_{\nu i}$ is the Hartree-Fock density matrix obtained from the converged SCF procedure. In order to take advantages from sparsity, the update of the $E_{\text{RI}}^{(2)}$ derivative is performed in the AO basis, previous back transformation of $P_{pq}^{(2)}$ and $W_{pq}^{(2)}$ from the MO basis.

In the framework of the GPW method, $h_{\mu\nu}^x$ contains the derivative of the matrix element of the electronic kinetic energy, short range part of the local pseudopotential and the non-local pseudopotential. These terms are computed analytically and explicit formulas can be found in Ref.[84]. The exact HF exchange contributions (last summation inside the squared brake in equation 20), are calculated consistently, via 4-index ERI derivatives, with the Γ-point implementation based on a short range (truncated) Coulomb operator in the case of PBC.[89,90] Due to the dual representation of the density in GPW, special care has to be taken for the derivative of the Hartree matrix elements (second term inside the squared brake in equation 20). In particular, it is convenient to reformulate the contribution coming from the Hartree

energy in terms of electrostatic densities, this is accomplished by exploiting the symmetry of $P^{\mathrm{HF}}_{\mu\nu}$, $P^{(2)}_{\mu\nu}$ and $(\mu\nu|\lambda\sigma)$ derivatives:

$$\begin{aligned}
&\sum_{\mu\nu\lambda\sigma} P^{(2)}_{\mu\nu} P^{\mathrm{HF}}_{\lambda\sigma} (\mu\nu|\lambda\sigma)^x = \\
&\quad = 2\sum_{\mu\nu} P^{(2)}_{\mu\nu} (\mu^x\nu| \sum_{\lambda\sigma} P^{\mathrm{HF}}_{\lambda\sigma} \phi_\lambda\phi_\sigma) + 2\sum_{\lambda\sigma} P^{\mathrm{HF}}_{\lambda\sigma} (\sum_{\mu\nu} P^{(2)}_{\mu\nu} \phi_\mu\phi_\nu|\lambda^x\sigma) \\
&\quad = 2\sum_{\mu\nu} P^{(2)}_{\mu\nu} (\mu^x\nu|\rho^{\mathrm{HF}}) + 2\sum_{\lambda\sigma} P^{\mathrm{HF}}_{\lambda\sigma} (\rho^{(2)}|\lambda^x\sigma) \\
&\quad = 2\sum_{\mu\nu} P^{(2)}_{\mu\nu} \int \phi^x_\mu(\vec{r})\phi_\nu(\vec{r}) v^{\mathrm{HF}}_H(\vec{r}) d\vec{r} + 2\sum_{\lambda\sigma} P^{\mathrm{HF}}_{\lambda\sigma} \int \phi^x_\lambda(\vec{r})\phi_\sigma(\vec{r}) v^{(2)}_H(\vec{r}) d\vec{r}
\end{aligned} \tag{22}$$

where the v^{HF}_H and $v^{(2)}_H$ are the Hartree potentials associated to $\rho^{\mathrm{HF}} = \sum_{\lambda\sigma} P^{\mathrm{HF}}_{\lambda\sigma} \phi_\lambda(\vec{r})\phi_\sigma(\vec{r})$ and $\rho^{(2)} = \sum_{\mu\nu} P^{(2)}_{\mu\nu} \phi_\mu(\vec{r})\phi_\nu(\vec{r})$. For the sake of completeness, it has to be mentioned that in the GPW scheme the total Hartree potential includes an additional term that comes from the introduction of a Gaussian charge distribution at each nucleus $\rho_c(\vec{r})$. This is commonly done in Ewald sum methods in order to decouple the long and short range treatment of the electrostatic interactions. The contribution to $E^{(2)^x}_{\mathrm{RI}}$ associated to this additional term is accounted by integrating the $\rho_c(\vec{r})$ derivative with the previously defined $v^{(2)}_H$ potential

$$2\int \rho^x_c(\vec{r}) v^{(2)}_H(\vec{r}) d\vec{r} \tag{23}$$

as likewise done for the similar term in standard HF method.[84] For efficiency reasons, in order to avoid the recomputations of integrals derivatives, the contraction of $P^{(2)}_{\mu\nu}$ and $W^{(2)}_{\mu\nu}$ is performed, when possible, at the same time with $P^{\mathrm{HF}}_{\mu\nu}$ and $W^{\mathrm{HF}}_{\mu\nu}$ matrices, *i.e.* simultaneously during the calculation of the HF energy derivatives.

At this stage, the only missing quantities that remain to be defined are $P^{(2)}_{pq}$ and $W^{(2)}_{pq}$. These matrices are usually calculated in the MO basis, and they are the result of the composition of terms that have a different definition according to which block of the matrix they refer, namely occupied-occupied (occ-occ), virtual-virtual (virt-virt) and occupied-virtual (occ-virt). Concerning the MP2 correction to the 1-PDM $P^{(2)}_{pq}$, the occ-occ and virt-virt

blocks are defined as:

$$P_{ij}^{(2)} = -\sum_{ab}^{virt}\sum_{k}^{occ} t_{ik}^{ab}\frac{(ja|kb)}{\epsilon_j + \epsilon_k - \epsilon_a - \epsilon_b} \tag{24}$$

$$P_{ab}^{(2)} = \sum_{ij}^{occ}\sum_{c}^{virt} t_{ij}^{ac}\frac{(ib|jc)}{\epsilon_i + \epsilon_j - \epsilon_b - \epsilon_c} \tag{25}$$

where t_{ij}^{ab} are the MP2 amplitudes, that in the restricted closed shell HF case take the form:

$$t_{ij}^{ab} = \frac{2(ia|jb) - (ib|ja)}{\epsilon_i + \epsilon_j - \epsilon_a - \epsilon_b}. \tag{26}$$

The virt-occ block of $P^{(2)}$ contains information related to the orbital relaxation caused by the perturbation x,[64] (*i.e.* first order response of the MO coefficients).[91] It is computed as the solution of the Z-vector equations[92]

$$\sum_{a}^{virt}\sum_{i}^{occ}\left[\delta_{ij}\delta_{ab}(\epsilon_a - \epsilon_i) + A_{aibj}\right]P_{ai}^{(2)} = -L_{bj} \tag{27}$$

where A_{aibj} is an element of the orbital Hessian matrix

$$A_{aibj} = 4(ai|bj) - (ab|ij) - (aj|bi), \tag{28}$$

and L is a specific RI-MP2 Lagrangian matrix given by:

$$L_{bj} = 2\sum_{a}^{virt}\sum_{Q}^{\mathrm{AUX}}(ba|Q)\Gamma_{ja}^{Q} - 2\sum_{i}^{occ}\sum_{Q}^{\mathrm{AUX}}(ij|Q)\Gamma_{ib}^{Q} + \sum_{ac}^{virt}P_{ac}^{(2)}A_{acbj} + \sum_{ik}^{occ}P_{ik}^{(2)}A_{ikbj}. \tag{29}$$

The first two terms in equation 29, namely $L_{bj}(1)$ and $L_{bj}(2)$, are computed within the mixed Lagrangian formalism,[70] that is, starting from the counterpart $L_{\mu j}(1)$ and $L_{b\nu}(2)$ in a mixed AO/MO basis

$$L_{jb}(1) = \sum_{\mu}^{\mathrm{AO}} C_{\mu b}L_{\mu j}(1) \tag{30}$$

$$L_{jb}(2) = \sum_{\nu}^{\mathrm{AO}} L_{b\nu}(2)C_{\nu j} \tag{31}$$

where:

$$L_{\mu j}(1) = 2\sum_{\nu}^{\text{AO}}\sum_{Q}^{\text{AUX}}(\mu\nu|Q)\Gamma^{Q}_{j\nu} \tag{32}$$

$$L_{b\nu}(2) = -2\sum_{i}^{occ}\sum_{Q}^{\text{AUX}}(i\nu|Q)\Gamma^{Q}_{ib}. \tag{33}$$

The above reformulation, allows to accumulate the contributions to $L_{\mu j}(1)$ and $L_{b\nu}(2)$ together with the contraction of $\Gamma^{Q}_{\mu\nu}$ and Γ^{PQ} with the integrals derivatives $(\mu\nu|Q)^x$ and $(P|Q)^x$. This choice is thus particularly convenient from a computational standpoint, since many intermediates, such as $\Gamma^{Q}_{j\nu}$, are in common for both updates and don't need to be recomputed. Moreover, the $(\mu\nu|Q)$ integrals have to be recalculated and this is performed at the same time with the computation of the corresponding derivatives, allowing a further saving of time since all grid operations, such as FFT's, are performed once for both terms.

The calculation of the off-diagonal elements of $P^{(2)}_{ij}$ and $P^{(2)}_{ab}$, defined in equation 24 and 25, can be equivalently computed (within a canonical reformulation[67,74,93]) from $L_{ij}(1) = \sum_{\mu} C_{\mu j} L_{\mu i}(1)$ and $L_{ab}(2) = \sum_{\nu} L_{a\nu}(2) C_{\nu b}$ as:

$$P^{(2)}_{ij} = \frac{1}{2}\frac{L_{ij}(1) - L_{ji}(1)}{\epsilon_j - \epsilon_i} \tag{34}$$

$$P^{(2)}_{ab} = -\frac{1}{2}\frac{L_{ab}(2) - L_{ba}(2)}{\epsilon_b - \epsilon_a}. \tag{35}$$

This choice suffers from numerical instability in the case $\epsilon_i \approx \epsilon_j$ or $\epsilon_a \approx \epsilon_b$, but, contrary to the case of equation 24 and 25, it offers a way for computing $P^{(2)}_{ij}$ and $P^{(2)}_{ab}$ that doesn't require $O(N^5)$ operations except for intermediates that are already available. Moreover, in a parallel implementation, where usually the work load is achieved by distributing independent ij pairs, the usage of equation 34 allows to drastically reduce the algorithmic complexity as well as avoiding the recomputation of MP2 amplitudes.[74]

Due to the large size of the orbital Hessian matrix $\mathbf{A}$ $(ov \times ov)$, the linear system of equation 27 is commonly solved by iterative techniques.[63,94–96] According to these methods, rather than calculating and storing the full $\mathbf{A}$, which is computationally inaccessible even for relatively small systems, at each iteration, the matrix-vector product $\sum_{ia} X_{ai} A_{aibj}$ is computed, with $\mathbf{X}$ being a trial solution. In this respect, it has to be noted that the orbital Hessian is made of a Coulomb part, first term in equation 28, and an Exchange part, last

two terms in equation 28.[91] These two updates of the matrix-vector product, have thus to be computed consistently to the way the Coulomb and Exchange contributions to the Fock matrix are calculated during the SCF procedure. In the actual case, this means that the former is obtained within the GPW scheme and the latter via 4-index ERIs. Again, for efficiency reasons, the AO representation is preferred so that sparsity can be exploited.

This leads to the following matrix-vector update for the Coulomb part:

$$\begin{aligned}
\sum_{ia} X_{ai}(ai|bj) &= \sum_{ai} X_{ai} \sum_{\mu\nu\lambda\sigma} C_{\mu a} C_{\nu i} (\mu\nu|\lambda\sigma) C_{\lambda b} C_{\sigma j} \\
&= \sum_{\mu\nu\lambda\sigma} \left[\sum_{ai} C_{\mu a} X_{ai} C_{\nu i} \right] (\mu\nu|\lambda\sigma) C_{\lambda b} C_{\sigma j} \\
&= \sum_{\mu\nu\lambda\sigma} Y_{\mu\nu} (\mu\nu|\lambda\sigma) C_{\lambda b} C_{\sigma j} \\
&= \sum_{\lambda\sigma} C_{\lambda b} C_{\sigma j} (\sum_{\mu\nu} Y_{\mu\nu} \phi_\mu \phi_\nu | \lambda\sigma) = \sum_{\lambda\sigma} C_{\lambda b} C_{\sigma j} (\rho^Y | \lambda\sigma) \\
&= \sum_{\lambda\sigma} C_{\lambda b} C_{\sigma j} \int \phi_\lambda(\vec{r}) \phi_\sigma(\vec{r}) v_H^Y(\vec{r}) d\vec{r}
\end{aligned} \tag{36}$$

where v_H^Y is the electrostatic potential obtained from the $\rho^Y(\vec{r}) = \sum_{\mu\nu}^{\text{AO}} Y_{\mu\nu} \phi_\mu(\vec{r}) \phi_\nu(\vec{r})$ density and $Y_{\mu\nu}$ is the back transformed matrix associated to the actual trial solution X_{ai}. The required update for the Exchange part via 4-index ERI[97] reads:

$$\begin{aligned}
-\sum_{ia} X_{ai}[(ab|ij) + (aj|bi)] &= \\
&= -\sum_{ai} X_{ai} \sum_{\mu\nu\lambda\sigma} C_{\mu a} C_{\nu i} [(\mu\lambda|\nu\sigma) + (\mu\sigma|\lambda\nu)] C_{\lambda b} C_{\sigma j} \\
&= -\sum_{\mu\nu\lambda\sigma} \left[\sum_{ai} C_{\mu a} X_{ai} C_{\nu i} \right] [(\mu\lambda|\nu\sigma) + (\mu\sigma|\lambda\nu)] C_{\lambda b} C_{\sigma j} \\
&= -\sum_{\mu\nu\lambda\sigma} Y_{\mu\nu} [(\mu\lambda|\nu\sigma) + (\mu\sigma|\lambda\nu)] C_{\lambda b} C_{\sigma j} \\
&= -\sum_{\mu\nu\lambda\sigma} C_{\lambda b} C_{\sigma j} (\mu\lambda|\nu\sigma) [Y_{\mu\nu} + Y_{\nu\mu}]
\end{aligned} \tag{37}$$

where the permutation symmetry of the AO-ERIs has been exploited.

Finally, the MP2 correction to the energy-weighted density matrix $W_{pq}^{(2)}$, is calculated as

follow: Occupied-occupied block:

$$W_{ij}^{(2)} = \frac{1}{2}\left\{W_{ij}^{(2)}[I] + W_{ij}^{(2)}[II] + W_{ij}^{(2)}[III]\right\} \tag{38}$$

$$W_{ij}^{(2)}[I] = 2\sum_{a}^{virt}\sum_{Q}^{\text{AUX}}(ja|Q)\Gamma_{ia}^{Q} = \sum_{\mu}^{\text{AO}} C_{\mu j} L_{\mu i}(1) \tag{39}$$

$$W_{ij}^{(2)}[II] = (\epsilon_i + \epsilon_j)P_{ij}^{(2)} \tag{40}$$

$$W_{ij}^{(2)}[III] = \sum_{pq}^{\text{MO}} P_{pq}^{(2)} A_{pqij} \tag{41}$$

Virtual-Virtual block:

$$W_{ab}^{(2)} = \frac{1}{2}\left\{W_{ab}^{(2)}[I] + W_{ab}^{(2)}[II]\right\} \tag{42}$$

$$W_{ab}^{(2)}[I] = 2\sum_{i}^{occ}\sum_{Q}^{\text{AUX}}(ib|Q)\Gamma_{ia}^{Q} = -\sum_{\nu}^{\text{AO}} C_{\nu b} L_{a\nu}(2) \tag{43}$$

$$W_{ab}^{(2)}[II] = (\epsilon_a + \epsilon_b)P_{ab}^{(2)} \tag{44}$$

Occupied-Virtual block:

$$W_{ai}^{(2)} = \frac{1}{2}\left\{W_{ai}^{(2)}[I] + W_{ai}^{(2)}[II]\right\} \tag{45}$$

$$W_{ai}^{(2)}[I] = 2\sum_{j}^{occ}\sum_{Q}^{\text{AUX}}(ji|Q)\Gamma_{ja}^{Q} = -\sum_{\nu}^{\text{AO}} C_{\nu i} L_{a\nu}(2) \tag{46}$$

$$W_{ai}^{(2)}[II] = 2\epsilon_i P_{ai}^{(2)} \tag{47}$$

The methodology presented up until here, is of general validity for any perturbation parameter x. In particular, for the calculation of the forces acting on the ions, the gradients of $E_{\text{RI}}^{(2)}$ with respect to the atomic positions have to be computed. Thus, within the GPW scheme, for which densities are represented in terms of both atom centered Gaussians and Plane Waves, only the derivatives of the former have to be considered since the latter are originless functions and do not depend of the atomic positions.

The RI-MP2 contribution to the total stress tensor is calculated according to:[98–100]

$$\Pi_{\alpha\beta}^{(2)} = -\frac{1}{3V}\sum_{\gamma=1}^{3}\frac{\partial E_{\text{RI}}^{(2)}}{\partial h_{\alpha\gamma}} h_{\gamma\beta}^{T} \tag{48}$$

where $h_{\alpha\gamma}$ are elements of the matrix of the cell vectors (Bravais lattice vectors) given by a_1, a_2 and a_3, that is $\mathbf{h} = [a_1, a_2, a_3]$. According to this, a change in $h_{\alpha\gamma}$ not only results in a scaling of all atom coordinates, but also affects the grid points over which the electrostatic densities and potentials are defined within the GPW scheme. The calculation of the integral derivatives in equation 13 has thus to account for this effect, resulting in additional terms to be considered. Note that the number of grid points is kept fixed in simulations employing a variable sell.

Again, the computation of $\mathbf{\Pi}^{(2)}$ can be split in two terms:

$$\mathbf{\Pi}^{(2)} = \mathbf{\Pi}^{(2)\text{-NS}} + \mathbf{\Pi}^{(2)\text{-S}} \tag{49}$$

that is, the non-separable part $\mathbf{\Pi}^{(2)\text{-NS}}$, specific of the RI-MP2 method, associated to the first two summations in equation 13, and the separable part $\mathbf{\Pi}^{(2)\text{-S}}$, giving the additional contribution derived from the contraction of the MP2 relaxed density matrices with the stress derivative of the Fock and Overlap matrix elements, last summation in equation 13.

Since in the non-separable part of ${E_{\text{RI}}^{(2)}}^x$, only integrals involving Coulomb interactions are required, $\mathbf{\Pi}^{(2)\text{-NS}}$ is obtained with a similar approach as that used for calculating the stress tensor of the Hartree energy,[101] for which the grid dependent contributions are evaluated following the work of Corso and Resta.[102] This leads respectively for the 3- and 2-center contributions ($\Pi_{\alpha\beta}^{(2)\text{-NS}} = 4\Pi_{\alpha\beta}^{(2)\text{-NS-3c}} - 2\Pi_{\alpha\beta}^{(2)\text{-NS-2c}}$) to:

$$\begin{aligned}
\Pi_{\alpha\beta}^{(2)\text{-NS-3c}} = -\frac{1}{3V}\Bigg[& \delta_{\alpha\beta}\left(\sum_{Q}^{\text{AUX}}\sum_{\mu\nu}^{\text{AO}}\Gamma_{\mu\nu}^{Q}(\mu\nu|Q)\right) \\
& + \sum_{Q}^{\text{AUX}}\sum_{\mu\nu}^{\text{AO}}\left[\Gamma_{\mu\nu}^{Q} + \Gamma_{\nu\mu}^{Q}\right]\int (R_{I\beta} - r_\beta)\nabla_{I\alpha}\phi_\mu(\vec{r})\phi_\nu(\vec{r})v_H^Q(\vec{r})d\vec{r} \\
& + \sum_{Q}^{\text{AUX}}\int (R_{I\beta} - r_\beta)\nabla_{I\alpha}\chi_Q(\vec{r})v_H^{\Gamma^Q}(\vec{r})d\vec{r} \\
& + \sum_{Q}^{\text{AUX}}\int\int \rho^Q(\vec{r})\rho^{\Gamma^Q}(\vec{r}')\frac{(r_\alpha - r'_\alpha)(r_\beta - r'_\beta)}{|\vec{r} - \vec{r}'|^3}d\vec{r}d\vec{r}'\Bigg]
\end{aligned} \tag{50}$$

$$\Pi_{\alpha\beta}^{\text{(2)-NS-2c}} = -\frac{1}{3V}\Bigg[\delta_{\alpha\beta}\left(\sum_{PQ}^{\text{AUX}}\Gamma^{PQ}(P|Q)\right) + 2\sum_{Q}^{\text{AUX}}\int(R_{I\beta}-r_\beta)\nabla_{I\alpha}\chi_Q(\vec{r})v_H^{\Gamma^Q}(\vec{r})d\vec{r} + \sum_{Q}^{\text{AUX}}\int\int\rho^Q(\vec{r})\rho^{\Gamma^Q}(\vec{r}')\frac{(r_\alpha-r'_\alpha)(r_\beta-r'_\beta)}{|\vec{r}-\vec{r}'|^3}d\vec{r}d\vec{r}'\Bigg] \tag{51}$$

where V is the volume of the cell, $\delta_{\alpha\beta}$ is the Kronecker delta, $\nabla_{I\alpha}$ is the α component of the gradient with respect to the atomic position and $R_{I\beta}$ refers to the β component of the atom coordinate. All other terms appearing in the above expressions have the same definitions given in equations 18 and 19, note that the quantities labeled with the Γ^Q superscript are computed differently for $\Pi_{\alpha\beta}^{\text{(2)-NS-3c}}$ and $\Pi_{\alpha\beta}^{\text{(2)-NS-2c}}$. In both cases the first term arises from the scaling of the system's volume while the last is associated with the derivative of the electrostatic potential v_H.[99] The remaining components are associated to the derivatives of the Gaussian basis functions.[103,104]

The RI-MP2 stress tensor is completed with the separable part $\mathbf{\Pi}^{\text{(2)-S}}$. This final update is performed together with the calculation of the stress components of the Hartree-Fock energy. The approach is relatively straightforward and is accomplished with a similar methodology as that one previously explained for the general derivative case. Again, special care has to be taken in the case of the Hartree energy term, for which additional terms arise due to the dual representation of the density in GPW. These additional contributions are obtained employing a similar approach as done in the case of the non-separable part but starting from equations 22 and 23.

In this section the general equations necessary for calculating the RI-MP2 energy derivatives have been presented with a particular focus on the way each term is calculated in the GPW framework. The presented approach has been applied for the calculation of the forces acting on the nuclei, and the stress tensor components. In summary, among all intermediates, only few quantities can be recognized as fundamental and need to be constructed in order to compute all the others, that is:

- $\Gamma_{ia}^{P}, \Gamma^{PQ} \rightarrow$ RI-MP2 non-separable correction to the 2-particle density matrix.
- $P_{ij}^{(2)}, P_{ab}^{(2)} \rightarrow$ Occupied-Occupied and Virtual-Virtual blocks of the MP2 correction to

the 1-particle density matrix.

- $L_{\mu j}(1), L_{b\nu}(2) \rightarrow$ Occupied and Virtual Lagrangian in the mixed AO-MO representation.

III. IMPLEMENTATION

The general flowchart of the algorithm for the calculation of the RI-MP2 energy derivatives can be summarized as follow:

1. Calculation of $(P|Q)$ and its inverse square root $V_{PQ}^{-1/2}$, subsequent evaluation of B_{ia}^{P}. These intermediates are evaluated within the RI-GPW approach.

2. Formation of the $(ia|jb)_{\mathrm{RI}}$ integrals from B_{ia}^{P} (equation 4), calculation of $E_{\text{RI-MP2}}$, assembly of Y_{ia}^{P} and $P_{ab}^{(2)}$ according to equations 16 and 25, evaluation of $P_{ij}^{(2)}$ via equation 24 for the diagonal and almost degenerate ($\epsilon_i \approx \epsilon_j$) elements.

3. Generation of $\Gamma_{ia}^{P}, \Gamma^{PQ}$ from Y_{ia}^{P}, B_{ia}^{P} and $V_{PQ}^{-1/2}$ (equations 15 and 17), evaluation of the non-separable contributions to $E_{\mathrm{RI}}^{(2)^x}$ and assembly of the $L_{\mu j}(1)$, $L_{b\nu}(2)$, occupied and virtual Lagrangian in the mixed AO-MO representation (equations 32 and 33).

4. Completion of $P_{ij}^{(2)}$ with $L_{ij}(1)$ for the non singular elements (equation 34). Construction of the RI-MP2 specific Lagrangian L_{bj} and solution of the Z-vector equations. Assembly of $P_{pq}^{(2)}$ and $W_{pq}^{(2)}$ and final evaluation of $E_{\mathrm{RI}}^{(2)^x}$ by contraction with $F_{pq}^{(x)}$ and $S_{pq}^{(x)}$.

The detailed description of each of these steps is reported in appendix A, with a particular focus on the parallelization strategy.

IV. BENCHMARK CALCULATIONS

A. Computational Details

The RI-GPW methods as implemented in CP2K[78] have been employed for all calculations in this manuscript. The correlation energy calculations are based on pseudopotentials of the form suggested by Goedecker, Teter and Hutter (GTH) in Ref.[88] but specifically parameterized for the methods employed to converge the wavefunction at the SCF level (HF or DFT). The same primary and auxiliary basis used in our previous works has been adopted.[58,59] These

are labeled as cc-DZVP, cc-TZVP and cc-QZVP, denoting double, triple and quadruple zeta quality respectively. They consist in valence-only correlation consistent type[105,106] basis sets, generated for being used with these pseudopotentials. The Hartee-Fock exchange calculations have been performed employing our robust Γ-point implementation[89,90] that allows for stable calculations in the condensed phase.[90,107] The Schwarz screening threshold for the HF calculations is in the range $10^{-8} - 10^{-10}$ for the energy, while for the related derivatives the threshold is in general relaxed by one order of magnitude. Periodic calculations require a truncated Coulomb operator,[90] using approximately half the length of the smallest edge of the simulation cell as truncation radius. The threshold for the SCF convergence was 10^{-6} or tighter. The PW cutoff for the HF and DFT (Perdew-Burke-Ernzerhof, PBE[108]) part of the calculations was $E_{cut} = 800$ and $E_{cut} = 1200$ Ry respectively to guarantee convergence of the exchange-correlation term. The correlation energy calculations employed high quality PW cutoffs of $E_{cut} = 300$ Ry, $E_{cut}^{rel} = 50$ Ry, $\epsilon_{filter} = 10^{-12}$, and $\epsilon_{grid} = 10^{-8}$,[58,59] unless mentioned otherwise. The convergence threshold for the Z-vector equations, measured as the norm of the residual vector, was 10^{-5} or tighter. Cluster boundary conditions have been adopted for solving the Poisson equation[87] in the case of gas phase systems, with cubic cells with edges between 15 and 20Å depending on molecule size. The pseudopotential and basis set parameters employed in this work can be found in Ref. [59].

For all the considered crystals, supercells have been generated by replicating the unit cell, so that the smallest edge was larger than 9Å, in order for the Γ-point approximation to be reasonable. The experimental geometries of the molecular crystals have been retrieved from the Cambridge Structural Database (CSD),[109] the structural data, together with the supercell used in the calculation and the CSD refcode, can be recovered from our previous works.[58,59] For both geometry and cell optimizations no symmetry constrains have been considered, and the latter have been performed under ambient pressure. The convergence thresholds have been set to 3.0×10^{-3} and 1.5×10^{-3} Bohr respectively for the maximum and root mean square (RMS) of the geometry changes, 4.5×10^{-4} and 3.0×10^{-4} Hartree·Bohr^{-1} respectively for the maximum and RMS of the forces, while a pressure tolerance of 100 bar has been considered for the cell optimization.

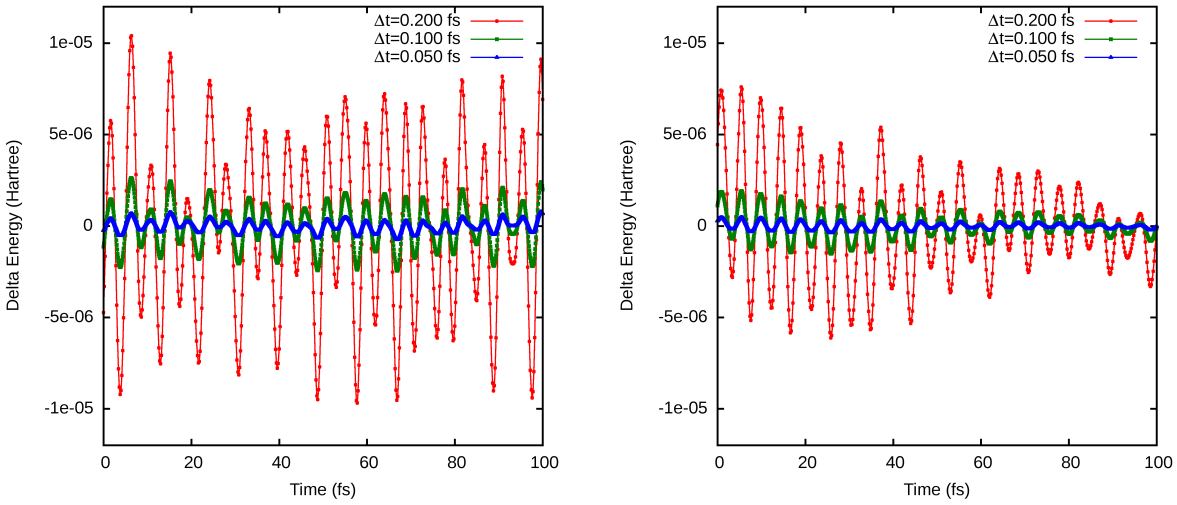
The counterpoise (CP) corrected cohesive energy per molecule at a given volume V and

for a given basis has been computed as[110,111]

$$E_{coh}^{CP}(V) = \frac{E_{supercell}(V)}{N_{mol}} - E_{mol}^{gas} - E_{mol+ghost}^{crystal}(V) + E_{mol}^{crystal}(V). \quad (52)$$

Here, N_{mol} is the number of molecules per supercell, $E_{supercell}(V)$ the total energy of the supercell, and $E_{mol+ghost}^{crystal}(V)$, $E_{mol}^{crystal}(V)$, and E_{mol}^{gas} the total energy of an isolated molecule in either the crystal geometry ($E_{mol+ghost}^{crystal}(V)$ and $E_{mol}^{crystal}(V)$) or a gas phase geometry (E_{mol}^{gas}). $E_{mol+ghost}^{crystal}(V)$, includes ghost atoms from the 12 nearest neighbor molecules for NH_3 and CO_2, and from the first coordination shell in all other cases. The gas phase geometries have been obtained by relaxation at the RI-MP2 level. Note that all geometry and cell relaxations have been performed without counterpoise correction.

B. Validation



(a) (b)

Figure 1. Energy fluctuation with respect to the average during a sequence of Born-Oppenheimer molecular dynamics simulations with periodic boundary conditions as a function of the time step Δt. The results are obtained employing the microcanonical ensemble (NVE) and the isobaric-isothermal ensemble (NpT) for (a) and (b) respectively. In both cases the system is made of 4 molecules of NH_3 in a cubic box with the cc-DZVP basis.

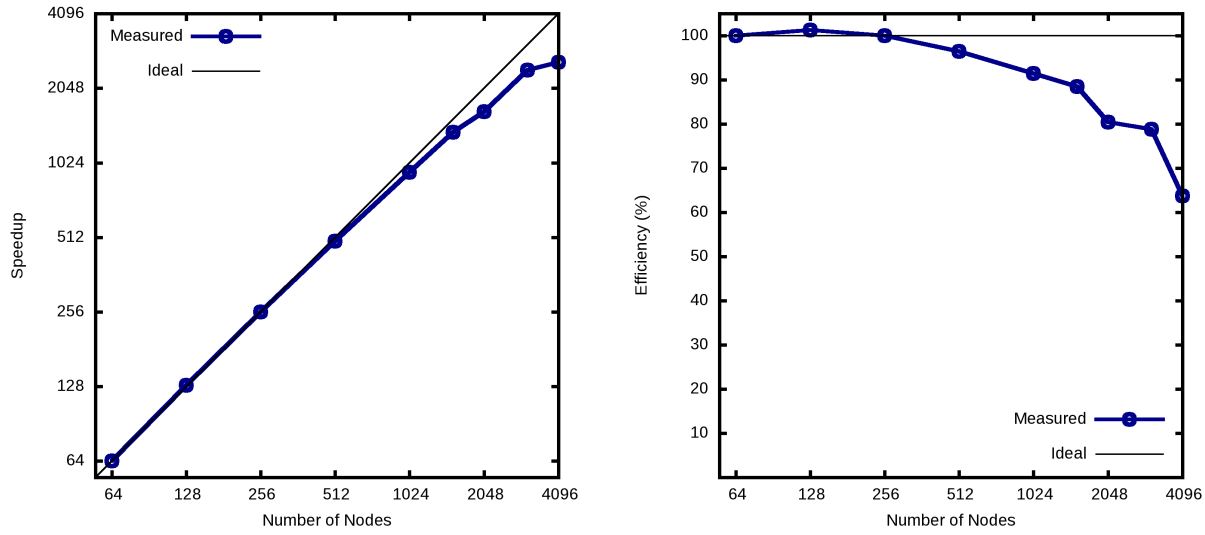
In order to validate that forces and stress are computed consistently to the way the RI-MP2 energy is calculated, a series of short Born-Oppenheimer molecular dynamics (BOMD)

simulations have been run with different time step Δt, employing the Velocity-Verlet algorithm for the integration of the equations of motion. The simulations have been performed in the microcanonical ensemble (NVE), figure 1a, and in the isobaric-isothermal ensemble (NpT), figure 1b. In the former only the forces acting on the atoms have to be computed while in the latter also the calculation of the stress tensor is required. The model system is made of 4 NH_3 in a cubic box with periodic boundary conditions and employing the cc-DZVP basis.

Within the Velocity-Verlet integration scheme, the total energy of an equilibrated system fluctuates around the average value with a standard deviation σ_E that is expected to be proportional to the square of the time step employed in the simulation, $\sigma_E \propto \Delta t^2$, meaning that, if Δt is halved, then σ_E is reduced roughly by a factor four. This is of course holds only in the case for which the forces, from which the accelerations are obtained, are computed as exact derivatives of the potential energy. The energies obtained from the BOMD trajectories are reported in figure 1, qualitatively showing that the magnitude of the fluctuations is roughly reduced by a factor four every time Δt is halved. More precisely the value of σ_E calculated for the NVE and NpT runs are 0.31, 1.2, 4.7 and 0.18, 0.72, 2.9 μHartree, respectively for time step of 0.05, 0.1 and 0.2 fs. These results are thus confirming the correctness of the RI-MP2 energy derivative implementation. We find that this approach is a stronger check than the mere comparison with numerical derivatives. For example, a large set of configurations are sampled making possible to track the propagation of possible small errors that may not be detected by numerical differentiation.

C. Performance of the Methods

The parallel performance of the algorithm for calculating the RI-MP2 energy gradients and stress has been measured for a system made of 64 water molecules in a cubic box with PBC at experimental density. The cc-TZVP basis has been employed resulting in 256 occupied orbitals, 3648 primary and 8704 auxiliary basis functions. The measured time includes all operations described in the previous sections, excluding only the contraction of $P^{(2)}_{pq}$ and $W^{(2)}_{pq}$ with the skeleton derivatives of the Fock and Overlap matrix elements, that is considered as part of the calculation of the HF energy derivatives. This means that also the solution of the Z-vector equation has been traced. In this respect, due to the limited amount of memory available for the smaller run, the AO-ERI's computed at the SCF level could not be kept in



(a) (b)

Figure 2. Speedup (a) and efficiency (b) with respect to 64 nodes for the calculation of the RI-MP2 energy gradients and stress of 64 bulk water molecules (cc-TZVP basis). Calculation performed on a CRAY-XC30 machine, each node consists of 8 processes.

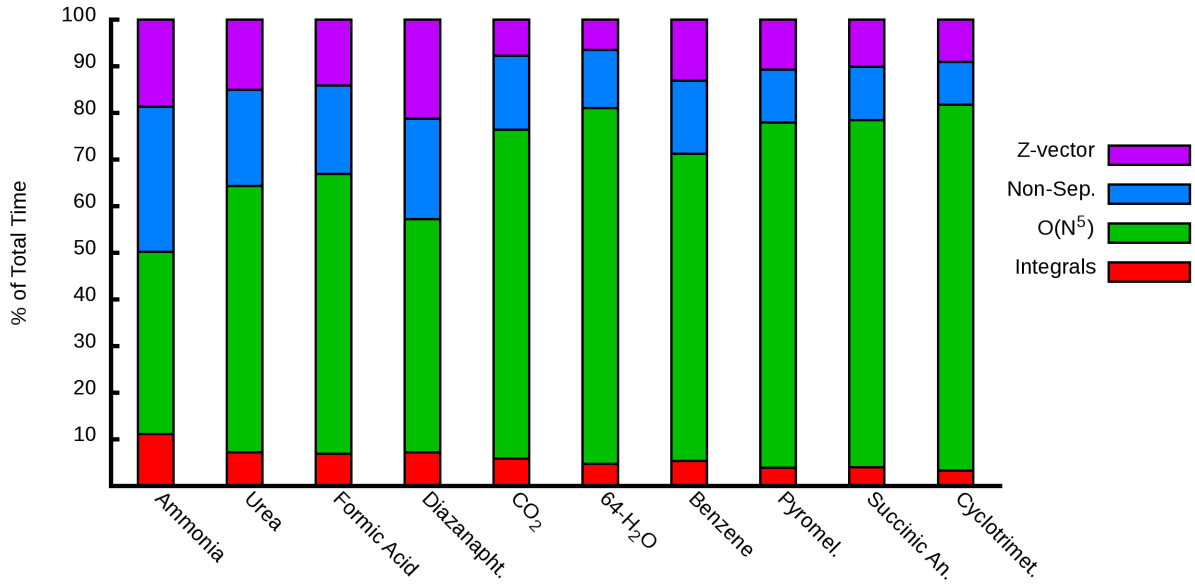


Figure 3. Relative time, express in term of percentage, spent in each of the most relevant part of the algorithm for the same benchmark calculations reported in table I. The meaning of the label in the legend stand for, Integrals = evaluation of $V_{PQ}^{-1/2}$ and B_{ia}^{P} (appendix A 1); $O(N^5)$ = evaluation of the $O(N^5)$ scaling intermediates (appendix A 2); Non-Sep. = evaluation of the non-separable contributions to $E_{\mathrm{RI}}^{(2)x}$ (appendix A 3); Z-vector = solution of the Z-vector equations, assembly of $P_{pq}^{(2)}$ and $W_{pq}^{(2)}$ (appendix A 4).

	o	n	N_a	t_{tot}	t_D	$\frac{t_D}{t_E}$	t_D^{GPU}	$\frac{t_D}{t_D^{\text{GPU}}}$
NH_3	128	2272	5312	3.15	1.53	4.20	1.47	1.04
U	192	2752	6784	5.97	3.58	4.59	2.89	1.24
FA	216	2760	6912	5.83	3.87	4.28	2.95	1.31
D	192	2992	7520	12.84	5.27	5.15	4.26	1.24
CO_2	256	2784	7296	7.94	4.99	4.15	3.50	1.43
H_2O	256	3648	8704	10.17	9.34	4.00	5.85	1.60
B	240	4128	10176	23.01	13.77	4.45	8.81	1.56
PD	312	3936	10208	28.96	17.48	4.13	9.80	1.78
SA	304	4144	10432	27.00	19.29	4.26	10.94	1.76
CT	336	4152	10560	29.71	22.30	4.16	11.97	1.86

Table I. Benchmark calculations for the RI-MP2 energy gradients and stress, time in min.@CRAY-XC30, 4096 processes, 512 GPU. U = Urea, B = Benzene, FA = Formic Acid, SA = Succinic Anhydride, D = 2,3-Diazanaphthalene, PD = Pyromellitic Dianhydride, CT = Cyclotrimethylene-Trinitramine, H_2O = 64 bulk water molecules, NH_3 = Ammonia Crystal (32 molecules), CO_2 = Carbon Dioxide Crystal (32 molecules). o, n and N_a represent the number of occupied orbitals, basis functions, auxiliary basis functions respectively. The reported timings represent: t_{tot} = Total time for computing HF and RI-MP2 energy, gradients and stress; t_D = time for computing RI-MP2 energy, gradients and stress; $\frac{t_D}{t_E}$ = ration between t_D and the time for computing only the RI-MP2 energy; t_D^{GPU} = the same as t_D but employing GPU; $\frac{t_D}{t_D^{\text{GPU}}}$ = observed speedup when using GPU.

core during the calculation of the RI-MP2 specific quantities, and their recomputation is thus necessary before solving the Z-vector equation.

The speedup and efficiency measured on a CRAY-XC30 machine are reported in figure 2. This machine mounts a GPU on each node, but for the actual measurements the usage of these devices has not been exploited. The algorithm displays good parallel scalability resulting in an efficiency higher than 80% for almost the whole range. At the full scale-out (32768 processes) the time for computing the RI-MP2 energy gradients and stress is 106 seconds. The relatively large drop in efficiency observed in going from 3072 to 4096 nodes is related to the scarce number of ij pairs processed by each MPI task in the latter, such that the time spent in computation becomes of the same order of the overheads related to communication.

In table I the timing for different benchmark calculations, obtained employing 512 nodes of a CRAY-XC30 machine, are reported, in this case also the impact of the usage of the GPU's has been considered. In general, for the actual implementation, the GPU's have been used to accelerate all the steps that are performed in the algorithm as matrix multiplication. This is of particular advantage for the RI-MP2 method since the expected most computationally intense part, *i.e.* the $O(N^5)$ steps, are all accomplished in this way.

For sake of completeness, in table I also the time $t_{\rm tot}$ necessary for the evaluation of the energy gradients and stress of the total energy (HF + RI-MP2) is reported. At the Hartree-Fock level the most expensive operations are related to the update of the Fock matrix with the exact exchange contributions, that involves the calculation of the AO-ERI's and relative derivatives. For the reported benchmarks, computing the derivatives of the RI-MP2 energy results in a percentage of the total time that grows systematically increasing the system size up to 75% for the largest case. The exception to that is the 64 bulk water case, for which the Schwarz screening, in particular for the $(\mu\nu|\lambda\sigma)$ derivatives, is particularly effective, resulting in a small time spent at HF level.

The total time necessary for calculating the RI-MP2 energy gradients and stress (t_D), reported in table I, is of the order of minutes for all the cases and results to be a factor between 4 to 5 times larger (t_D/t_E) than what required for the calculation of the RI-MP2 energy only. According to the analysis done in appendix A 2 in the limit of very large system, *i.e.* when the $O(N^5)$ steps are by far the most time consuming part of the total computation, the ratio t_D/t_E is expected to be between 3 to 4. This is just the theoretical limit, since the calculation of the integral derivatives as well as the solution of the Z-vector equations will always give a non negligible overhead to the calculation.

The relative time spent in each part of the algorithm, for the different benchmark calculations, are reported in figure 3. For all cases, except the Ammonia crystal, the time spent for calculating the $O(N^5)$ intermediates is more that 50% of the total time, reaching almost 80% for the largest case (CT). The computation of the RI specific quantities, $V_{PQ}^{-1/2}$ and B_{ia}^{P} is the cheapest operation requiring less than 10% of the total computational effort for all calculations. The evaluation of the non-separable contributions to $E_{\rm RI}^{(2)^x}$ is dominated by the calculation of the 3-center integrals and associated derivatives, and results to be roughly a factor 3 more expensive than the computation of $V_{PQ}^{-1/2}$ and B_{ia}^{P}. In this respect, the computation of the non-separable contributions to the RI-MP2 stress tensor takes around

30% of time spent in this part while the rest is related to the evaluation of the forces. The remaining part is associated with the solution of the Z-vector equations that can require a variable percentage of the overall time according the the number of iterations necessary to reach convergence. For all the reported cases, the amount of memory was enough for keeping in core the AO-ERI's computed at the SCF level during the evaluation of the RI-MP2 specific quantities. This allowed to avoid their recomputation for the solution of the Z-vector, making this operation less computationally demanding.

Finally the time for the calculation of the RI-MP2 energy gradients and stress by exploiting the GPU's for the operations performed as matrix multiplication is reported in table I labeled as t_D^{GPU} together with the observed speedup compared to the CPU only case (t_D/t_D^{GPU}). As shown in the table, the speedup is modest for the smaller cases while approaching a factor 2 for the larger ones. Focusing on the largest benchmark calculation (CT), the observed speedup for the overall calculation is roughly 1.9, while specifically for the steps performed as matrix multiplications the observed speedups are in general greater than 4.

D. System Size Scaling

In order to verify the cost models presented in the previous sections, the time for each of the individual steps of the algorithm has been measured for increasing system sizes. The test system is based on a supercell containing 32 water molecules with a cc-DZVP basis set that has been replicated in one dimension up to 5 times. The results are reported in figure 4 where the obtained timings have been fitted with the function $y = bx^a$, yielding the measured scaling exponent a associated with each different step.

In figure 4 the labels "Integrals" and "Non-Sep." refer to all operations described in appendix A 1 and A 3 respectively. The evaluation of the $O(N^5)$ scaling intermediates (appendix A 2) have been traced in more details, reporting separately the measured timings for the generation of the $(ia|jb)_{\mathrm{RI}}$ integrals, the update of Y_{ia}^P and $P_{ab}^{(2)}$, and communication. Note that, the solution of the Z-vector equations in this case, due the combination of the system topology and small basis, takes a negligible time with respect to the total time and thus has not been reported in the plot.

The observed scaling for "Integrals" and "Non-Sep." is 2.1 and 2.3 respectively showing that the integration of the electrostatic potential over pairs of basis elements $\mu\nu$ is the

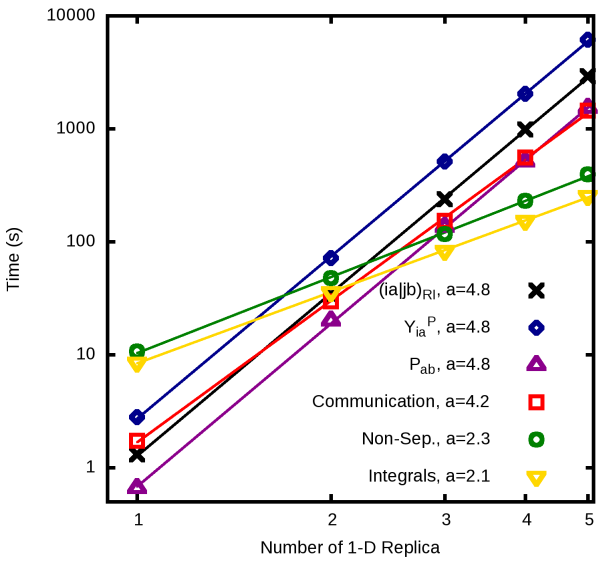


Figure 4. Time spent in the various significant part of the algorithm for the calculation of the RI-MP2 energy gradients, as a function of the number of replicas of the supercell, containing 32, 64, 96, 128 and 160 molecules of H_2O respectively (cc-DZVP basis). Timing measured on a CRAY-XK7 machine employing 2400 processes without GPU's. Lines represent a linear two-parameter fit of the form $y = bx^a$. The values of a for each operation are reported in the legend.

dominant part within the tested sizes. This operation is in fact expected to scale as $O(N^2)$, while the additional parts, such as integral transformation, scaling as $O(N^4)$, make the exponent a slightly larger than 2. This effect is more pronounced for the latter compared to the former due to the higher number of $O(N^4)$ steps performed in the update of the non-separable part of $E_{\mathrm{RI}}^{(2)x}$.

For the evaluation of the $O(N^5)$ intermediates ($(ia|jb)_{\mathrm{RI}}$, Y_{ia}^P and $P_{ab}^{(2)}$) the observed scaling is in all cases 4.8, while communication has a measured a of 4.2. This is in agreement with the performance models derived for these operations, that is 5 for the former and 4 for the latter. From the comparison of the timings of the individual $O(N^5)$ steps, it is observed that the update of Y_{ia}^P takes roughly a factor 2 more compared to the generation of $(ia|jb)_{\mathrm{RI}}$, while the latter results two times more expensive than the update of $P_{ab}^{(2)}$. The reason for the first observation is related to the fact that, for a given ij pair, the update of Y_{ia}^P and the generation of $(ia|jb)_{\mathrm{RI}}$ require for both $O(v^2 N_a)$ operations, but for the former (Y_{ia}^P) this is performed 2 times (for i and j respectively). The update of $P_{ab}^{(2)}$ requires $O(2v^3)$ for each ij and since in this case $N_a \simeq 4v$ the observed time scales as $O(v^2 N_a/2)$ that is half than what

is needed for generating $(ia|jb)_{\mathrm{RI}}$.

E. Applications

1. *Solid NH_3 and CO_2*

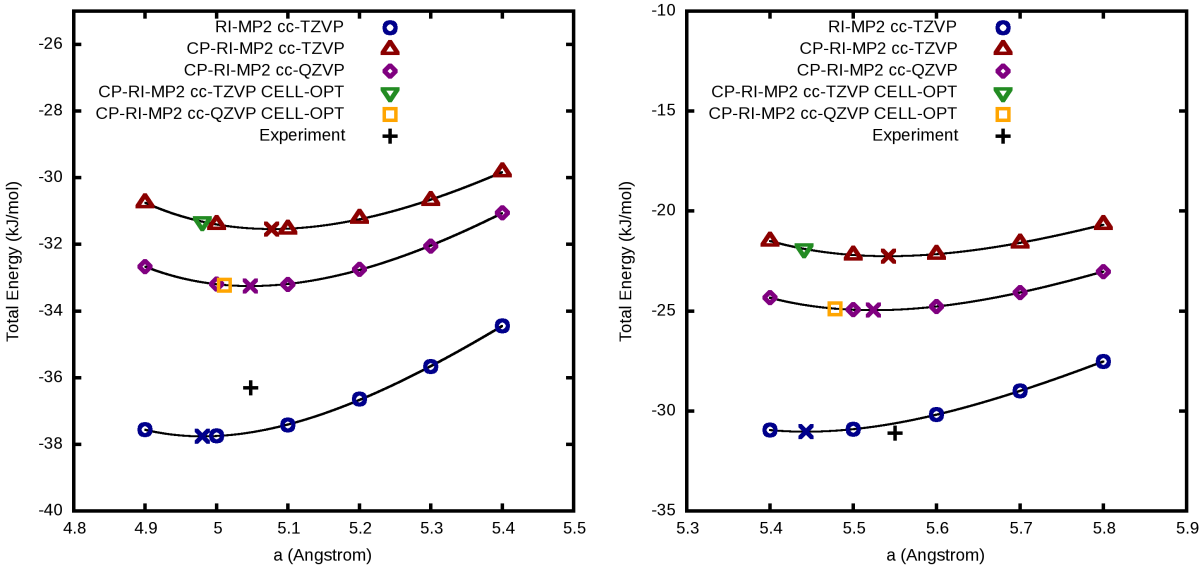


Figure 5. Location of the minima for NH_3 (a) and CO_2 (b), computed at the RI-MP2 level of theory with different basis sets obtained with different approaches. The lattice parameter optimization curves have been fitted with a third order Birch-Murnagham equation, the crosses represent the location of the minimum point for each curve. CP means that the cohesive energy have been counterpoise corrected.

Ammonia and carbon dioxide molecular crystals represent two simple benchmark systems useful for judging the performance of a method. The dominant interactions in the two cases are very different in nature, being weak hydrogen bond for NH_3 and purely van der Waals for CO_2. These systems have been extensively investigated both experimentally and theoretically. Concerning the theoretical studies, many of them are MP2 theory based methods such as periodic-canonical MP2,[58] periodic-local MP2 (LMP2),[110] incrementally corrected LMP2,[118] embedded many-body expansion[119,120] and hybrid QM/MM fragment method.[121]

Two approaches can be used for calculating the equilibrium lattice parameter (a) and cohesive energy (E_{coh}) of these crystals. The first approach (direct method) is to perform

	NH_3				CO_2		
	a	$r_{\text{N-H}}$	$\angle_{\text{H-N-H}}$	E_{coh}	a	$r_{\text{C-O}}$	E_{coh}
Opt-CP (PBE-D3)	5.00	1.027	107.5	-44.97	5.73	1.171	-26.50
Fit-Opt	4.98	-	-	-37.76	5.44	-	-31.04
Fit-Opt-CP	5.08	-	-	-31.54	5.54	-	-22.26
Fit-Opt-CP (QZ)	5.05	-	-	-33.25	5.52	-	-24.96
Opt-CP	4.98	1.017	107.4	-31.32	5.44	1.166	-21.90
Opt-CP (QZ)	5.01	1.017	107.2	-33.23	5.48	1.168	-24.88
Exp.	5.048	1.01 - 1.06	107.5	-36.3	5.55 - 5.62	1.155 - 1.12	-31.1

Table II. Equilibrium cohesive energy (E_{coh} in kJ/mol per molecule) and structural properties (lattice parameter a and bond length in Å, angles in degree) for the NH_3 and CO_2 crystals calculated employing different methods. Except when specified otherwise, the basis set is cc-TZVP and the level of theory RI-MP2 (QZ stands for cc-QZVP basis). CP means that E_{coh} is counterpoise corrected, Opt denotes full cell optimization while Fit-Opt refers to the results obtained by fitting the curves shown in figure 5 (third order Birch-Murnagham equation). Experimental values from Ref.[110] (see also Ref.[112–114]) and Ref.[115–117] and references therein for NH_3 and CO_2 respectively.

a cell optimization followed by the calculation of E_{coh} for the equilibrium structure. The second one (indirect method) consists in the optimization of the geometry at various fixed volumes from which the equilibrium quantity a and E_{coh} are derived by fitting employing, *e.g.* a third order Birch-Murnagham equation. The former is computationally more efficient since a single optimization has to be carried out, moreover it allows to gain more information on the local molecular structure at equilibrium. On the other hand, this approach suffers from basis set superposition error (BSSE) that can be particularly large within MP2 theory. This drawback can be remedied by increasing the basis set, or, in the indirect method, by considering counterpoise corrected energies. Both approaches have been considered employing the cc-TZVP and cc-QZVP basis, the computed equilibrium properties are summarized in table II.

As shown in figure 5, at the triple zeta level, the lattice parameter optimization curves are calculated both with and without CP correction. The position of the obtained minima, indicated by the crosses in the plots, shows clearly the large discrepancies between the two approaches. As expected, for both systems, without CP correction the crystals result overbind with shorter equilibrium lattice parameter and larger cohesive energy. The CP correction fixes this issue giving values for a closer to the experimental one but E_{coh} in general higher.

The cell relaxation provides converged structures that preserve the cubic symmetry of the crystals within the numerical accuracy of the method. With the cc-TZVP basis, the obtained lattice parameters from cell optimization (Opt) are essentially the same as those obtained from the non-CP curve optimizations (Fit-Opt) and thus substantially shorter than those evaluated with the CP correction (Fit-Opt-CP). This divergence is effectively reduce when using the cc-QZVP basis, showing a clear trend in the convergence. In fact, by inspection of figure 5, it can be seen that the Fit-Opt-CP approach tends to converge, with respect to the basis set, from larger values of a opposite to the case of the cell optimization.

At the quadruple zeta level the lattice parameters obtained from the cell optimization are 5.01 and 5.48 Å respectively for NH_3 and CO_2, with associated cohesive energies of -33.2 and -24.9 kJ/mol. With the same basis the CP curve optimization procedure, Fit-Opt-CP(QZ), gives similar results for E_{coh}, but slightly larger values of a being respectively 5.05 and 5.52 Å. From the observations previously stated it can be concluded that, for the supercell considered in this work, the complete basis set limit for the equilibrium lattice parameter should be within these values, that is between 5.01-5.05 and 5.48-5.52 Å, respectively for NH_3 and CO_2.

For ammonia, good agreement is found with the values reported by Maschio *et al.*[110] obtained with the aug(d,f)-TZPP basis, that is $a = 5.02$ Å and $E_{coh} = -36.6$ kJ/mol, while a larger deviation in the lattice parameter is observed for carbon dioxide ($a = 5.59$ Å and $E_{coh} = -26.6$ kJ/mol). A better agreement for the lattice constant a of the CO_2 crystal is obtained when comparing with the value of 5.52 and 5.46 Å reported respectively by Bygrave *et al.*[119] and Sode *et al.*,[120] calculated with a CP augmented triple-zeta basis for the former and augmented quadruple-zeta basis for the latter. In this case also the optimized C-O bond length match the values reported by these authors.

As a comparison, also the results obtained with the PBE functional including the Grimme D3[122] correction have been reported. For ammonia good agreement is found between RI-MP2 and PBE-D3 in the lattice constant, while the cohesive energies display a large discrepancy. On the other hand, for CO_2 the E_{coh} is estimated roughly the same but the value of a is around 5% larger than that calculated with RI-MP2.

Since many effects, such as temperature dependence and zero-point vibrational energies, are neglected, caution has to be exercised when comparing the obtained results with experiments. For both crystals, at the QZ level, the lattice parameters are less than 2% shorter than the experimental ones while the cohesive energies display larger deviations.

A result of the analysis reported here is that, due to the slow convergence of the MP2 energy with respect to basis set size, also the bulk structures are subject to BSSE. Here, in order to remedy this issue, counterpoise correction and larger basis sets have been used. An alternative which requires significant additional development, and is thus not tested here, is to use explicitly correlated treatments such as F12-MP2,[31,32] which is also possible in the condensed phase.[33] In addition, in previous work,[58,59] the effect of the supercell size on the cohesive energy has been studied in detail, showing that an L^{-3} extrapolation, with L the edge of the unit cell, can be used for converge the system size effect inherent to the used Γ-point approximation.

2. Molecular Crystals

Geometry and cell optimization at the RI-MP2 level employing the cc-TZVP basis have been carried out for a set of molecular crystals. This set includes the crystal of Urea (U), Formic Acid (FA), Benzene (B), Pyromellitic Dianhydride (PD), Succinic Anhydride (SA) and Cyclotrimethylene-Trinitramine (CT). The dominant intermolecular interactions for these crystals cover a large range, from hydrogen-bond to dipole-dipole to purely van der Waals. Also the size of the molecules across the investigated set are quite different going from a minimum of 5 (FA) up to 21 atoms for the largest case (CT). For the relaxed structures the counterpoise corrected cohesive energy has been computed, the obtained results, compared with the experimental values, are reported in table III.

For all cases, the cell optimization preserves the orthorhombic symmetry of the crystals, roughly keeping the experimental aspect ratio. The obtained lattice parameter are in all cases underestimated compared to experiment, with deviations ranging from 1 up to 7%. This leads to a much larger discrepancies for the cell volumes for which a maximum deviation of 15% is observed for the Benzene crystal. A slightly smaller error, around 8% for the volume, is observed for the Urea and Formic Acid crystals, cases for which the intermolecular interactions are mainly of dipole-dipole and hydrogen-bond types. This difference can be rationalized by inspection of the S22 set,[123] for which it is shown that MP2 is in general giving poor performance for the complexes with predominant dispersion contribution, such as the case of Benzene, while the the results are usually better for hydrogen bonded complexes.[123,124]

As in the cases of solid NH_3 and CO_2, the cell optimization at the triple-zeta level is

	Geo-Opt	Cell-Opt			Exp.		
	E_{coh}	abc	V	E_{coh}	abc	V	E_{coh}
		5.45			5.65		
U	-97.1	5.45	138	-96.6	5.65	150	-92
		4.64			4.70		
		10.06			10.24		
FA	-55.6	3.36	179	-54.9	3.54	194	-68
		5.31			5.36		
		7.14			7.40		
B	-58.8	8.78	400	-63.0	9.44	473	-45
		6.39			6.78		
		10.16			10.79		
PD	-125.7	10.18	754	-130.4	10.79	863	-83
		7.29			7.41		
		5.14			5.43		
SA	-82.6	6.57	382	-84.8	6.97	443	-81
		11.31			11.72		
		12.88			13.18		
CT	-116.6	11.05	1452	-115.8	11.57	1634	-112
		10.21			10.71		

Table III. Counterpoise corrected cohesive energy E_{coh} (kJ/mol) equilibrium volume V (Å^3) and lattice parameters abc (Å) for different molecular crystals calculated after structural relaxation at the RI-MP2 level of theory employing the cc-TZVP basis. The meaning of the labels are: U = Urea, FA = Formic Acid, B = Benzene, PD = Pyromellitic Dianhydride, SA = Succinic Anhydride, CT = Cyclotrimethylene-Trinitramine. Geo-Opt refers to geometry optimization at experimental volume while Cell-Opt stands for full cell relaxation. The experimental E_{coh} are obtained from sublimation enthalpies $\Delta H(s)$ with opposite sign. For the experimental date see Ref [58,111], as well as http://webbook.nist.gov/chemistry/ and Cambridge Structural Database.[109]

clearly affected by the BSSE. In order to estimate to which extent the observed error in the converged lattice parameters is due to BSSE or to intrinsic limitation of the MP2 theory, a full cell relaxation employing the cc-QZVP has been performed for the Formic Acid crystal. The obtained lattice constants at the quadruple-zeta level are $a = 10.20$, $b = 3.41$ and $c = 5.33$ Angstrom while the computed cohesive energy is -59.3 kJ/mol. Increasing the basis set is thus reducing the error in the equilibrium volume to 4.2%. In particular, while for the a and c vectors the agreement is fairly good, a large deviation is observed for the b

lattice parameter, such that the discrepancy in the volume is almost completely determined by this underestimation. Solid formic acid consists of infinite chains of molecules linked by hydrogen bonds, while inter-chain interactions are dominated by dispersion and weak C-H$\cdots$O contacts. The weak intermolecular interactions act along the cell vectors a and b, while the hydrogen bonded formic acid molecules form infinite chains approximately oriented along cell vector c.[125,126] The compression observed along the b vector can be interpreted as the result of the overbinding of dispersion interactions at the MP2 level. Note that the cell optimization for FA at the PBE level with the cc-TZVP basis gives $a = 11.19$, $b = 4.20$ and $c = 5.24$ Å, in good agreement with $a = 10.91$, $b = 4.11$ and $c = 5.28$ Å, reported by Tosoni *et al.*[125] obtained employing the Ahlrichs' TZP basis.

The CP corrected cohesive energies E_{coh} reported in table III have been computed for the relaxed structures obtained after geometry optimization at experimental volume and cell optimization. Even if a strong structural relaxation take place in cell optimization, this doesn't reflect into a large variation of E_{coh} compared to that obtained after geometry optimization at fixed experimental volume. This is a direct consequence of the weak binding interactions that dominate in these systems, giving rise to relatively flat potential energy surfaces with respect to the cell parameters. Moreover, the shrinking of the volume during the cell relaxation, leads to structures of the individual molecules that are usually less stable when extracted from the relaxed geometries obtained from Cell-Opt than those obtained from Geo-Opt. This effect partially compensates the gain in energy due the optimization of the cell parameters.

When comparing E_{coh} with experimental sublimation enthalpies $\Delta H(s)$, it can be noted that at the MP2 level good agreement is found when the crystal are bound with mixed electrostatic-dispersion interactions such in the case of Urea, Succinic Anhydride and Cyclotrimethylene-Trinitramine.[127] For crystals such as Benzene and Pyromellitic Dianhydride, purely bounded with van der Waals dispersion interactions, large deviations are observed, resulting in general in an overestimation of the cohesive energy.

V. CONCLUSIONS

In this work, the equations for calculating the derivatives of the MP2 energy in the framework of the Resolution of Identity Gaussian and Plane Waves method have been derived

and presented in detail. The central aspect in the computation of the derivatives of the correlation energy within the RI-GPW approach is the dual representation of the RI fitting density in term of Gaussian and Plane Waves auxiliary functions. The latter representation is equivalent to expressing the electrostatic densities over regular grids in space. This allows the straightforward conversion of these quantities into the associated potentials by solving the Poison equation in G space and exploiting Fourier transformations for switching between direct and reciprocal representations. In this way the evaluation of the integral derivatives is accomplished consistently to the way the energy is calculated. This approach is of general validity and it has been applied to the calculation of the forces acting on the atoms (gradients) as well as for the derivative with respect to the cell volume (stress tensor).

For the presented scheme a massively parallel algorithm has been designed displaying, with respect of the system size, cubic, quartic and quintic requirements respectively for the memory, communication and computation. All these requirements scale with increasing number of processes. The implementation is based on a hybrid OpenMP/MPI scheme for which the parallelization is achieved by distributing the work over subgroups of processes rather than over a single task. This allowed to achieved a more flexible memory management and reduced communication without loss of computational efficiency. The measured performance displays excellent parallel scalability and efficiency up to thousands of nodes. Moreover, in the actual implementation the computationally more demanding part, that is the $O(N^5)$ steps, are accelerated by employing GPU's showing a gain of almost a factor two compared to the standard CPU only case for large systems.

Several benchmark calculations have been reported for validating the theoretical and methodological aspects. It is shown that the presented scheme is efficient, accurate and robust especially for systems in the condensed phase. The effort for the calculation of the derivatives at the RI-MP2 level is between 4 to 5 times more expensive than computing only the energy. Geometry optimization as well as full cell relaxation have been performed for a variety of molecular crystals. The obtained results are in general good agreement with both previously reported calculations and experimental data. Furthermore, the actual implementation allows to fully exploit the computational power of new generation supercomputers, such that the derivatives of the RI-MP2 energy can be performed within minutes for systems containing hundreds of atoms and thousands of basis functions. This opens the possibility to perform structural relaxation or even molecular dynamics at the MP2 level of theory for condensed

phase systems with accurate basis sets, and our recent study on the relative stability of the different phases of ice XV[128] is a case in point.

The methodology presented here is a general framework that can be extended for the calculation of the energy derivatives evaluated at the double-hybrid density functional[129–131] and random-phase approximation[132] level, for which (N^4) scaling implementations have been reported.

VI. ACKNOWLEDGMENTS

J.V. acknowledges financial support by the European Union FP7 in the form of an ERC Starting Grant under contract no. 277910. We acknowledge that the results of this research have been achieved using the PRACE Research Infrastructure resource Hermit based in Germany at Stuttgart (HLRS). Additional calculations were enabled by the Swiss National Supercomputer Centre (CSCS) under project ID ch5. The research leading to these results has received funding from the Swiss University Conference through the High Performance and High Productivity Computing (HP2C) Programme and the Platform for Advanced Scientific Computing (PASC) programme.

Appendix A: Parallel Algorithm

The parallel algorithm described here is a scalable implementation of the RI-MP2 energy derivatives displaying, with respect of the system size, cubic, quartic and quintic requirements respectively for the memory, communication and computation.

1. Evaluation of $V_{PQ}^{-1/2}$ and B_{ia}^{P} with the RI-GPW Method

The parallel algorithm for the evaluation of $V_{PQ}^{-1/2}$ and B_{ia}^{P} within the RI-GPW approach has been presented in great details in Ref.[59]. Here only the most important features of the parallel implementation are recalled in order to help the description of the algorithm in the next sections.

The parallelization strategy is based on a two level distribution of the workload, obtained by splitting the total N_p processes available into N_G groups, consisting of N_w processes ($N_p = N_G N_w$). The first level is associated to the work performed for a single given auxiliary

basis function χ_P or vector $|P) = \sum_Q \chi_Q(\vec{r}) V_{QP}^{-1/2}$. The parallelization at this level is obtained within the N_w processes of each group based on a hybrid OpenMP/MPI scheme involving, for example, parallel FFTs, halo-exchanges, numerical integration of the basis functions over the electrostatic potential and sparse matrix multiplications. The second level corresponds to a distribution of these nearly independent calculations among the different groups. This is achieved by splitting the total number of auxiliary basis function N_a into N_G ranges $[P_{start}^{n_P}, P_{end}^{n_P}]$, each of them labeled with a given n_P coordinate, and assigned to the corresponding group. Additionally, each of the N_w processes within a group is given an index n_w, so that a processes is uniquely identified by its coordinate (n_P, n_w). Finally, the a virtual index is split in N_w ranges $[a_{start}^{n_w}, a_{end}^{n_w}]$, while the index i is kept over the full set of occupied orbitals.

The workload distribution described so far allows for a scalable parallel implementation for the integral computation. In fact, the communication intense steps are restricted to within the group, made of a small number of tasks, while each group works independently for the different χ_P or $|P)$ associated to its range $[P_{start}^{n_P}, P_{end}^{n_P}]$.

Focusing on the calculation of B_{ia}^P, for each $P \in [P_{start}^{n_P}, P_{end}^{n_P}]$ the computational procedure can be summarized as follow:

- Evaluation of the density $\rho^P(\vec{r}) = \sum_Q \chi_Q(\vec{r}) V_{QP}^{-1/2}$ on the real space grid.
- Calculation of the electrostatic potential $v_H^P(\vec{r})$ associated to $\rho^P(\vec{r})$. This is obtained by first transferring $\rho^P(\vec{r})$ from the real to reciprocal space via FFT, solving the Poisson equation in Fourier space and finally back transferring the potential, with an additional FFT, from reciprocal to real space.
- Integration of the potential over the pairs of basis set elements, $B_{\mu\nu}^P = \int \phi_\mu(\vec{r}) \phi_\nu(\vec{r}) v_H^P(\vec{r}) d\vec{r}$.
- Transformation of $B_{\mu\nu}^P$ from the AO to the MO basis by two consecutive matrix-matrix multiplication, that is $B_{i\nu}^P = \sum_\mu C_{\mu i} B_{\mu\nu}^P$ and finally $B_{ia}^P = \sum_\nu B_{i\nu}^P C_{\nu a}$.

The asymptotically dominating part of this procedure is associated to the last index transformation that has a formal scaling of $O(ovnN_a/N_p)$ while the integration of the potential has a cost that grows only quadratically with the system size. Nevertheless, due to the small prefactor associated to the former, the latter is usually more computationally demanding, even for relatively large systems.[59]

At the end of this step each process stores the elements of B_{ia}^{P} for all i, $P \in [P_{start}^{n_P}, P_{end}^{n_P}]$ and $a \in [a_{start}^{n_w}, a_{end}^{n_w}]$.

2. Evaluation of the $O(N^5)$ Scaling Intermediates

With $O(N^5)$ scaling intermediates all the quantities that require for their construction a quintic computational effort are implied. Within the RI-MP2 method these intermediates are $(ia|jb)_{\mathrm{RI}}$, Y_{ia}^{P}, $P_{ab}^{(2)}$ and $P_{ij}^{(2)}$ evaluated respectively with equations 4, 16, 25 and 24 for which the formal computational effort grows as $O(o^2v^2N_a)$, $O(o^2v^2N_a)$, $O(o^2v^3)$ and $O(o^3v^2)$. The efficient construction of these intermediates is of prime importance since they are the asymptotically most expensive calculations of the RI-MP2 energy derivatives evaluation.

While in a serial algorithm their computation is relatively straightforward, a parallel implementation has to face the problems connected with the distributed storage of the precursors, such as B_{ia}^{P}, as well as the balancing of the workload over processes. Regarding the latter issue, a distribution of independent ij pairs ($i \leq j$) is of particular convenience for the evaluation of $(ia|jb)_{\mathrm{RI}}$, Y_{ia}^{P} and $P_{ab}^{(2)}$, but not for the $P_{ij}^{(2)}$, for which the distribution of the ab pairs would be preferred. In order to overcome this complication, while retaining the ease of the ij distribution as well as avoiding additional events of communication, the $P_{ij}^{(2)}$ are evaluated via equation 24 only for the diagonal and almost degenerate elements ($|\epsilon_i - \epsilon_j| < t_{\mathrm{sing}}$) while employing equation 34 for all the others. This approach is referred as semi-canonical and t_{sing} is a threshold for discriminating which of the ij pairs have to be treated as almost degenerate, expected anyway to be a small fraction of the total.

The pseudocode for the parallel evaluation of $(ia|jb)_{\mathrm{RI}}$, Y_{ia}^{P}, $P_{ab}^{(2)}$ and the diagonal elements $P_{ii}^{(2)}$ is sketched in figure 6, while the update of $P_{ij}^{(2)}$ for the almost degenerate pairs is shown in figure 7. The completion of $P_{ij}^{(2)}$ for the remaining elements is performed later, when $L_{\mu j}(1)$ is made available. In the pseudocodes the expected computational effort, expressed in term of order of, is reported for the most important operations.

The algorithm shown in figure 6 closely resembles the structure employed for the RI-MP2 energy evaluation described in Ref.[59]. As mentioned in the previous section the B_{ia}^{P} intermediate is distributed such that each process stores elements for all occupied i, $P \in [P_{start}^{n_P}, P_{end}^{n_P}]$ and $a \in [a_{start}^{n_w}, a_{end}^{n_w}]$. That is, the virtual index a is distributed over a small number of MPI tasks within the group G while the auxiliary index P is distributed over the

According to available memory define batch size B_S and size N_r of replication group (R)

Split the N_G groups into N_R subgroups ($N_R = N_G/N_r$)

Assign to each group G its coordinate in the subgroup n_R

Define the set $\{n_{R,w}\}$ including all processes having the same coordinate (n_R, n_w)

Define new set $\{P_{n_R,n_w}\}$ given as the union of $[P^{n'_P}_{start}, P^{n'_P}_{end}]$, $n'_P \in \{n_{R,w}\}$

Create β^P_{ia} by collecting B^P_{ia}, $P \in \{P_{n_R,n_w}\}$ from all other processes in $\{n_{R,w}\}$

Assign to each group G its set $\{IJ\}_G$ of IJ batches, ($I \leq J$, size of I and J given by B_S)

Loop over IJ batches ($IJ \in \{IJ\}_G$)

Collect $A^i_{aP} = \beta^P_{ia}$ and $E^j_{aP} = \beta^P_{ja}$, ($i \in I$, $j \in J$, all P, $a \in [a^{n_w}_{start}, a^{n_w}_{end}]$)

from all other processes in R with my same n_w coordinate $o^2 v N_a/(N_p B_S)$ (1)

Loop over ij ($i \in I$, $j \in J$)

Loop over n'_w, $n'_w \in G$

Collect E^j_{bP}, ($b \in [a^{n'_w}_{start}, a^{n'_w}_{end}]$)

$I_{ab} = (ia|jb) = \sum_P A^i_{aP} E^j_{bP}$, ($a \in [a^{n_w}_{start}, a^{n_w}_{end}], b \in [a^{n'_w}_{start}, a^{n'_w}_{end}]$) $o^2 v^2 N_a/N_p$ (2)

End n'_w Loop

$t_{ab} = (2I_{ab} - I_{ba})/\Delta^{ab}_{ij}$ $(*)$

$E^{(2)} = E^{(2)} + (2 - \delta_{ij}) \sum_{ab} I_{ab} t_{ab}$, ($a \in [a^{n_w}_{start}, a^{n_w}_{end}]$, all b) $o^2 v^2/N_p$ (3)

$P^{(2)}_{ab} = P^{(2)}_{ab} + \sum_c t_{ca} I_{cb}/\Delta^{cb}_{ij}$, ($a \in [a^{n_w}_{start}, a^{n_w}_{end}]$, all b) $(*)$ $o^2 v^3/N_p$ (4)

$P^{(2)}_{ii} = P^{(2)}_{ii} - \sum_{ab} t_{ab} I_{ab}/\Delta^{ab}_{ij}$, ($a \in [a^{n_w}_{start}, a^{n_w}_{end}]$, all b) $o^2 v^2/N_p$ (5)

$\Xi^i_{aP} = \Xi^i_{aP} + \sum_c t_{ac} E^j_{cP}$, ($a \in [a^{n_w}_{start}, a^{n_w}_{end}]$, all P, actual ij) $(*)$ $o^2 v^2 N_a/N_p$ (6)

if $i \neq j$ then

$P^{(2)}_{ab} = \sum_c t_{ac} I_{bc}/\Delta^{bc}_{ij}$, ($a \in [a^{n_w}_{start}, a^{n_w}_{end}]$, all b) $(*)$ $o^2 v^3/N_p$ (7)

$P^{(2)}_{jj} = P^{(2)}_{jj} - \sum_{ab} t_{ab} I_{ab}/\Delta^{ab}_{ij}$, ($a \in [a^{n_w}_{start}, a^{n_w}_{end}]$, all b) $o^2 v^2/N_p$ (8)

$\Lambda^j_{aP} = \Lambda^j_{aP} + \sum_c t_{ca} A^i_{cP}$, ($a \in [a^{n_w}_{start}, a^{n_w}_{end}]$, all P, actual ij) $(*)$ $o^2 v^2 N_a/N_p$ (9)

end if

End ij Loop

Collect Ξ^i_{aP} and Λ^j_{aP} from all other processes in R with my same n_w coordinate: $o^2 v N_a/(N_p B_S)$ (10)

Accumulate $X^P_{ia} = X^P_{ia} + \Xi^i_{aP}$, if $i \neq j$ also $X^P_{ja} = X^P_{ja} + \Lambda^j_{aP}$ $o^2 v N_a/N_p$ (11)

(all ij processed in this cycle by all $G' \in R$, $P \in \{P_{n_R,n_w}\}$, $a \in [a^{n_w}_{start}, a^{n_w}_{end}]$)

End IJ Loop

Collect X^P_{ia} from all processes in $\{n_{R,w}\}$: $Y^P_{ia} = Y^P_{ia} + X^P_{ia}$, (all i, $P \in [P^{n_P}_{start}, P^{n_P}_{end}]$, $a \in [a^{n_w}_{start}, a^{n_w}_{end}]$)

Global summation of $P^{(2)}_{ab}$ across all process having the same coordinate n_w

Global summation of $E^{(2)}$ and diagonal elements $P^{(2)}_{ii}$ across all process

Figure 6. Pseudocode of the parallel algorithm for computing $E^{(2)}$, $P^{(2)}_{ab}$, Y^P_{ia} and the diagonal elements of $P^{(2)}_{ij}$. All statements labeled with $(*)$ involves in-group communication, these operations are shown explicitly in the algorithm only for the calculation of I_{ab}, in all other cases the procedure is performed likewise.

Compute the number d of ij pairs ($i<j$) for which $|\epsilon_i-\epsilon_j|<t_{\rm sing}$

Define the set $\{ij\}_{\rm sing}$ of size d containing the almost degenerate ij pairs

Assign to each group G its set $\{ijk\}_G$ of ijk triplets, ($ij\in\{ij\}_{\rm sing}$, all occupied k)

Loop over ijk triplets ($ijk\in\{ijk\}_G$)

Collect $A^i_{aP}=\beta^P_{ia}$, $E^j_{aP}=\beta^P_{ja}$ and $U^k_{aP}=\beta^P_{ka}$ (actual ijk, all P, $a\in[a^{n_w}_{start},a^{n_w}_{end}]$)

from all other processes in R with my same n_w coordinate $dovN_a/N_p$ (1)

Loop over n'_w, $n'_w\in G$

Collect U^k_{bP}, ($b\in[a^{n'_w}_{start},a^{n'_w}_{end}]$)

$I^{ik}_{ab}=(ia|kb)=\sum_P A^i_{aP}U^k_{bP}$, ($a\in[a^{n_w}_{start},a^{n_w}_{end}], b\in[a^{n'_w}_{start},a^{n'_w}_{end}]$) dov^2N_a/N_p (2)

$I^{jk}_{ab}=(ja|kb)=\sum_P E^j_{aP}U^k_{bP}$, ($a\in[a^{n_w}_{start},a^{n_w}_{end}], b\in[a^{n'_w}_{start},a^{n'_w}_{end}]$) dov^2N_a/N_p (3)

End n'_w Loop

$t^{ik}_{ab}=(2I^{ik}_{ab}-I^{ik}_{ba})/\Delta^{ab}_{ik}$ $\quad(*)$

$P_{\rm val}=-\sum_{ab}t^{ik}_{ab}I^{jk}_{ab}/\Delta^{ab}_{jk}$, ($a\in[a^{n_w}_{start},a^{n_w}_{end}]$, all b) dov^2/N_p (4)

$P^{(2)}_{ij}=P^{(2)}_{ij}+P_{\rm val}$

$P^{(2)}_{ji}=P^{(2)}_{ji}+P_{\rm val}$

End ijk Loop

Global summation of $P^{(2)}_{ij}$ ($ij\in\{ij\}_{\rm sing}$) across all process

Figure 7. Pseudocode of the parallel algorithm for computing the off diagonal elements of $P^{(2)}_{ij}$ for almost degenerate ij pairs ($\epsilon_i\approx\epsilon_j$). The meaning of the symbols is the same as in figure 6.

large amount of N_G groups. The same kind of distribution has been adopted for Y^P_{ia} while $P^{(2)}_{ab}$ and $P^{(2)}_{ij}$ are created in a replicated form within each group, only the virtual index a of the former is distributed over the elements of the group with the usual $a\in[a^{n_w}_{start},a^{n_w}_{end}]$.

The parallelization is achieved by distributing independent ij pairs over the N_G groups. Each group, for a given ij pair, perform the following set of operations:

1. B^P_{ia} is redistributed such that the full range of the auxiliary index P, for i and j, is collected on local buffers from all other groups, while keeping the virtual index distribution, $a\in[a^{n_w}_{start},a^{n_w}_{end}]$

2. the $(ia|jb)_{\rm RI}$ integrals are generated for the actual ij pair in a matrix-multiplication fashion (equation 4)

3. the amplitudes t^{ab}_{ij} are formed from $(ia|jb)_{\rm RI}$ and $\Delta^{ab}_{ij}=\epsilon_i+\epsilon_j-\epsilon_a-\epsilon_b$

4. the contributions to $P^{(2)}_{ab}$ and $P^{(2)}_{ii}$ are accumulated into the relative local buffers

5. the contribution to Y^P_{ia} are accumulated into two intermediates namely Ξ^i_{aP} and Λ^j_{aP}

distributed within the group such that each process stores the full auxiliary index P and $a \in [a_{start}^{n_w}, a_{end}^{n_w}]$

6. Ξ_{aP}^{i} and Λ_{aP}^{j} are redistributed over all groups and accumulated into the local Y_{ia}^{P}.

In the above procedure, the first and the last steps involves inter-group communication. They can be considered isomorphic with the difference that in the former, for the actual ij pair, each process collect the full range of auxiliary functions for $a \in [a_{start}^{n_w}, a_{end}^{n_w}]$, in the latter, each task collects all the ij indices processed by all other groups for its preassigned range of $P \in [P_{start}^{n_P}, P_{end}^{n_P}]$ and $a \in [a_{start}^{n_w}, a_{end}^{n_w}]$. All other operations are performed locally within the group, that is, only a small amount of communication takes place restricted to the members of the group.

The main source of communication of the parallel algorithm is thus related to the inter-group redistribution steps mentioned above. According to the implementation designed for the RI-MP2 energy, three key aspects are considered in order to enhance the efficiency of these operations:

- B_{ia}^{P} is replicated into β_{ia}^{P} distributed within subgroups (the replication groups R). This allows to restrict the number of processes that have to communicate at each cycle.

- The messages are exchanged employing point-to-point communication. This allows to avoid global synchronization of processes while keeping a low memory usage.

- The ij pairs are communicated in batches, reducing the overall number of messages.

While the first point is more a technical ingredient, the other two are mandatory in order to obtain a scalable implementation, so that the required communication is reduced increasing the number of processes.

In more details, following the pseudocode given in figure 6, as a first stage, according to the available memory the size N_r of the replication groups R is defined and the elements of B_{ia}^{P} are replicated into β_{ia}^{P} distributed among the members of R. N_R being the number of replication groups, a ratio N_R/N_G of ~ 0.1, has been shown to be a good compromise between the time necessary for the replication and the gain in communication in the subsequent steps. The remaining memory available per process is then used to define the maximum batch size B_S and the total number of IJ batches ($I \leq J$) are then distributed statically over the N_G

groups. To achieve the best possible load balance, the number of IJ batches is restricted to be a multiple of the number of groups N_G and the remaining ij single pairs are again statically distributed over groups.

At this point each group loops over its preassigned IJ batches and, as a first task, collect from all other members of the replication group R, the elements $A^i_{aP} = \beta^P_{ia}$ and $E^j_{aP} = \beta^P_{ja}$. For all ij pairs in the actual IJ batch the operations between 2 and 5 previously described are performed. This allows to calculate the contribution coming from the actual ij to $P^{(2)}_{ab}$,$P^{(2)}_{ii}$,$P^{(2)}_{jj}$ and Ξ^i_{aP}, Λ^j_{aP}. Before moving to the next batch, the second inter-group communication step take place, redistributing Ξ^i_{aP} and Λ^j_{aP} within the members of the replication group. This corresponds in the pseudocode to lines 10 and 11, for which the intermediate X^P_{ia} is introduced. This quantity stays to Y^P_{ia} as β^P_{ja} stays to B^P_{ja}, *i.e.* X^P_{ia} represents the replicated version of Y^P_{ia} collecting the contributions coming from all the ij pairs processed by the groups G in the replication group R. For this reason, at the end of the loop over IJ batches, an additional step of decomposition of X^P_{ia} is required in order to generate Y^P_{ia} in its final form, that is, distributed such that each process stores the elements for all occupied i, $P \in [P^{n_P}_{start}, P^{n_P}_{end}]$ and $a \in [a^{n_w}_{start}, a^{n_w}_{end}]$. The two final steps are global summation of the elements of $P^{(2)}_{ab}$ and diagonal of $P^{(2)}_{ij}$, with the difference that the latter is summed over all processes while the former only across those that share the same virtual index range $a \in [a^{n_w}_{start}, a^{n_w}_{end}]$, *i.e* those labeled with the same coordinate n_w.

At this stage, what remains to be done is the update of the elements of $P^{(2)}_{ij}$ for the potentially singular ij pairs. This is accomplished first by checking the total number d of almost degenerate ij pairs, *i.e.* the pairs for which $|\epsilon_i - \epsilon_j| < t_{\text{sing}}$. For each of these a loop over all occupied k is performed, the $(ia|kb)_{\text{RI}}$ and $(ja|kb)_{\text{RI}}$ are generated, and the $P^{(2)}_{ij}$ element is updated according to equation 24.

As shown by the pseudocode in figure 7, the parallelization of these steps is obtained in a very similar way as done for the computation of $P^{(2)}_{ab}$, with the main difference that in this case the ijk triplets are distributed over the N_G groups. The number of ijk triplets $(d \times o)$ is usually small compared to the total number of ij pairs (o^2), for this reason, the communication scheme employing batches is not exploited since it may lead to poor balance of the work load. As in the previous case, the procedure is finalized with a global summation over all processes of the almost degenerate elements of $P^{(2)}_{ij}$. It has to be noted that, in the case that no potentially singular ij pairs are detected (as in most of the cases), this part of

the algorithm is completely skipped.

As a summary, the parallel algorithm described in this section can be spit into two relevant parts: communication and computation of the $O(N^5)$ scaling intermediates. The first part has a cost that can be estimated to be $O(o^2 v N_a/(N_p B_S))$, that is derived by considering that the total number of messages exchanged by each process is $O(o^2/(B_S^2 N_G))$ while the time required for each event of communication (considered to be proportional to the message sizes) is $O(v N_a B_S/N_w)$. This implies that communication is eventually an $O(N^4)$ operation whose effort scales with the number of processes $N_p = N_w N_G$. It has to be noted that, compared to the RI-MP2 energy algorithm, this operation is expected to be roughly two times more expensive since it involves not only the redistribution of β_{ia}^P, line 1 in the pseudocode 6, but also the similar operation for Ξ_{aP}^i and Λ_{aP}^j, line 10. Concerning the computation of the $O(N^5)$ intermediates, these are reported in the pseudocode 6 at line 2, the generation of the $(ia|jb)_{\mathrm{RI}}$ integrals, at line 4 and 7, update of $P_{ab}^{(2)}$ and at line 6 and 9, update of Ξ_{aP}^i and Λ_{aP}^j. Again, a comparison with the energy RI-MP2 algorithm, for which only the generation of the $(ia|jb)_{\mathrm{RI}}$ is required, leads to the conclusion that, for the actual implementation, the $O(N^5)$ part is expected to be roughly 3 to 4 times more expensive. As an additional remark, all these tasks are accomplished as matrix multiplications, and thus the performance of highly optimized routines, such as DGEMM, can be exploited as well as accelerated by employing a hybrid implementation that utilizes graphics processing units (GPUs).

3. Non-Separable Contribution to $E_{\mathbf{RI}}^{(2)^x}$, Assembly of $L_{\mu j}(1)$ and $L_{b\nu}(2)$

The calculation of the non-separable contribution to $E_{\mathrm{RI}}^{(2)^x}$, $L_{\mu j}(1)$ and $L_{b\nu}(2)$ is performed within the same procedure since many intermediates, deriving for example from the back-transformation of Γ_{ia}^P, are in common between these evaluations and can be computed within a loop over the auxiliary basis function index P. In general, the non-separable contributions to $E_{\mathrm{RI}}^{(2)^x}$ are calculated by contraction of 3- and 2-center RI integral derivatives with the non-separable correction to the 2-particle density matrix, computed in the framework of the RI-GPW approach (equations 18 and 19); while $L_{\mu j}(1)$ and $L_{b\nu}(2)$ are evaluated with equations 32 and 33. The pseudocode of this procedure is reported in figure 8.

Prior the calculation of these quantities it is thus necessary to assembly Γ_{ia}^P and Γ^{PQ} (3- and 2-index non-separable correction to the 2-PDM). This is achieved from the B_{ia}^P, Y_{ia}^P and

Redistribute Y_{ia}^P and B_{ia}^P into $(ov \times N_a)$ parallel distributed matrices $Y_{ia,P}$, $B_{ia,P}$

Create $(ov \times N_a)$ parallel distributed matrix $\Gamma_{ia,P}$

Create $(N_a \times N_a)$ parallel distributed matrices $\gamma_{P,Q}$ and $\Gamma_{P,Q}$

Perform parallel matrix-matrix multiplication:

$\Gamma_{ia,P} = \sum_Q Y_{ia,Q} V_{QP}^{-1/2}$ ovN_a^2/N_p (1)

$\gamma_{P,Q} = \sum_{ia} \Gamma_{ia,P} B_{ia,Q}$ ovN_a^2/N_p (2)

$\Gamma_{P,Q} = \sum_R \gamma_{P,R} V_{RQ}^{-1/2}$ N_a^3/N_p (3)

Redistribute $\Gamma_{P,Q} \rightarrow \Gamma^{PQ}$, (all P, $Q \in [P_{start}^{n_P}, P_{end}^{n_P}]$)

Redistribute $\Gamma_{ia,P} \rightarrow \Gamma_{ia}^P$, ($P \in [P_{start}^{n_P}, P_{end}^{n_P}]$, for each group all i, a

stored in the form of a parallel distributed matrix defined within the group)

Create $L_{\mu i}^G(1)$ and $L_{a\nu}^G(2)$ parallel distributed matrices defined within the group

Loop over P auxiliary basis functions ($P_{start}^{n_P} \leq P \leq P_{end}^{n_P}$)

Calculate density $\rho^P(\vec{r}) = \chi_P(\vec{r})$ on the real space grid $N_a S/N_p$ (4)

Transfer $\rho^P(\vec{r}) \rightarrow \rho^P(\vec{G})$: $\rho^P(\vec{G}) = \mathbf{FFT}[\rho^P(\vec{r})]$ $N_a S \log(S)/N_p$ (5)

Solve Poisson's Equation: $\rho^P(\vec{G}) \rightarrow v_H^P(\vec{G})$ $N_a S/N_p$ (6)

Transfer $v_H^P(\vec{G}) \rightarrow v_H^P(\vec{r})$: $v_H^P(\vec{r}) = \mathbf{FFT^{-1}}[v_H^P(\vec{G})]$ $N_a S \log(S)/N_p$ (7)

Calculate integral derivatives: $D_Q^P = 2 \int \chi_Q^x(\vec{r}) v_H^P(\vec{r}) d\vec{r}$ (all AUX Q) N_a^2/N_p (8)

$E_{\mathrm{RI}}^{(2)^x} = E_{\mathrm{RI}}^{(2)^x} - 2 \sum_Q \Gamma^{PQ} D_Q^P$, (all Q, actual P)

Back-transform Γ_{ia}^P, virtual index: $\Gamma_{i\nu}^P = \sum_a \Gamma_{ia}^P C_{\nu a}$ $ovnN_a/N_p$ (9)

Back-transform $\Gamma_{i\nu}^P$, occupied index and symmetrize: $\Gamma_{\mu\nu}^P = \sum_i C_{\mu i} \Gamma_{i\nu}^P$ $on^2 N_a/N_p$ (10)

Calculate integrals and derivatives:

$I_{\mu\nu}^P = \int \phi_\mu(\vec{r}) \phi_\nu v_H^P(\vec{r}) d\vec{r}$ (all $\mu\nu$) nN_a/N_p (11)

$D_{\mu\nu}^P = 2 \int \phi_\mu^x(\vec{r}) \phi_\nu v_H^P(\vec{r}) d\vec{r}$ (all $\mu\nu$) nN_a/N_p (12)

$E_{\mathrm{RI}}^{(2)^x} = E_{\mathrm{RI}}^{(2)^x} + 4 \sum_{\mu\nu} \Gamma_{\mu\nu}^P D_{\mu\nu}^P$, (all $\mu\nu$, actual P)

Transform first index of $I_{\mu\nu}^P$ to occupied MO: $I_{i\nu}^P = \sum_\mu C_{\mu i} I_{\mu\nu}^P$ onN_a/N_p (13)

Accumulate contribution to the Lagrangian in the mixed AO-MO representation:

$L_{\mu i}^G(1) = L_{\mu i}^G(1) + 2 \sum_\nu I_{\mu\nu}^P \Gamma_{i\nu}^P$ onN_a/N_p (14)

$L_{a\nu}^G(2) = L_{a\nu}^G(2) - 2 \sum_i I_{i\nu}^P \Gamma_{ia}^P$ $ovnN_a/N_p$ (15)

Calculate density $\rho^{\Gamma^P}(\vec{r}) = \sum_{\mu\nu} \Gamma_{\mu\nu}^P \phi_\mu(\vec{r}) \phi_\nu(\vec{r})$ on the real space grid $N_a S/N_p$ (16)

Transfer $\rho^{\Gamma^P}(\vec{r}) \rightarrow \rho^{\Gamma^P}(\vec{G})$: $\rho^{\Gamma^P}(\vec{G}) = \mathbf{FFT}[\rho^{\Gamma^P}(\vec{r})]$ $N_a S \log(S)/N_p$ (17)

Solve Poisson's Equation: $\rho^{\Gamma^P}(\vec{G}) \rightarrow v_H^{\Gamma^P}(\vec{G})$ $N_a S/N_p$ (18)

Transfer $v_H^{\Gamma^P}(\vec{G}) \rightarrow v_H^{\Gamma^P}(\vec{r})$: $v_H^{\Gamma^P}(\vec{r}) = \mathbf{FFT^{-1}}[v_H^{\Gamma^P}(\vec{G})]$ $N_a S \log(S)/N_p$ (19)

$E_{\mathrm{RI}}^{(2)^x} = E_{\mathrm{RI}}^{(2)^x} + 4 \int \chi_P^x(\vec{r}) v_H^{\Gamma^P}(\vec{r}) d\vec{r}$ (actual P) N_a/N_p (20)

End P Loop

Create $L_{\mu i}(1)$ and $L_{a\nu}(2)$ parallel distributed matrices defined aver all processes

Redistribute $L_{\mu i}^G(1), L_{a\nu}^G(2) \rightarrow L_{\mu i}(1), L_{a\nu}(2)$ accumulating contributions coming from other processes

Figure 8. Pseudocode of the parallel algorithm for computing the mixed AO-MO Lagrangian $L_{\mu i}(1)$, $L_{a\nu}(2)$ and the non-separable contribution to $E_{\mathrm{RI}}^{(2)^x}$ with the RI-GPW approach.

$V_{PQ}^{-1/2}$ intermediates by a sequence of parallel matrix multiplications (lines 1, 2 and 3 in figure 8). Since both B_{ia}^{P} and Y_{ia}^{P} are distributed such that each process stores the elements for all i, $P \in [P_{start}^{n_P}, P_{end}^{n_P}]$ and $a \in [a_{start}^{n_w}, a_{end}^{n_w}]$, a redistribution step is require in order to reorganize the data into the form of a parallel distributed matrix suitable for performing efficiently the multiplications. A second redistribution step is then performed for the computed Γ_{ia}^{P} and Γ^{QP} such that each group G stores for $P \in [P_{start}^{n_P}, P_{end}^{n_P}]$ all ia of the former, and all Q of the latter. The elements of Γ_{ia}^{P} reshuffled in this way are further organized within the group in the form of a parallel distributed matrix.

At this point the actual computation of the non-separable part of $E_{\mathrm{RI}}^{(2)^x}$ as well as $L_{\mu j}(1)$ and $L_{b\nu}(2)$ is performed by accumulating the contributions to these quantities associated to each auxiliary element P. As done for the calculation of B_{ia}^{P}, the parallelization is achieved by letting each group G work on its preassigned range of $P \in [P_{start}^{n_P}, P_{end}^{n_P}]$. Again all operations associated to a given P are performed in parallel within the members of the groups

The series of required operations for each P are shown in details in the pseudocode reported in figure 8. As a first step, according to the RI-GPW scheme, the electrostatic potential $v_H^P(\vec{r})$ associated to the single auxiliary basis function $\chi_P(\vec{r})$ is evaluated and made available on the real space grid. The potential is thus integrated over the auxiliary basis function derivatives $\chi_Q^x(\vec{r})$ for all Q and subsequently contracted with the relative elements of Γ^{QP} giving the non-separable contribution to $E_{\mathrm{RI}}^{(2)^x}$ from the 2-center ERI's.

At this point two steps of back-transformation of Γ_{ia}^{P} are performed obtaining both $\Gamma_{i\nu}^{P}$ and $\Gamma_{\mu\nu}^{P}$. The previously calculated potential $v_H^P(\vec{r})$ is now integrated over the pair of primary basis functions $\phi_\mu(\vec{r})\phi_\nu(\vec{r}) \rightarrow I_{\mu\nu}^{P}$ and associated derivatives $\phi_\mu^x(\vec{r})\phi_\nu(\vec{r}) \rightarrow D_{\mu\nu}^{P}$. The integral derivatives $D_{\mu\nu}^{P}$ are contracted with the fully back-transformed $\Gamma_{\mu\nu}^{P}$ giving the first non-separable contribution to $E_{\mathrm{RI}}^{(2)^x}$ from the 3-center ERI's, while the plain integrals $I_{\mu\nu}^{P}$ are multiplied with $\Gamma_{i\nu}^{P}$ and accumulated into a local buffer $L_{\mu j}^{G}(1)$ of $L_{\mu j}(1)$. This update can exploit the sparsity of $I_{\mu\nu}^{P}$ making this step $O(on)$ for each P. The contribution to $L_{b\nu}(2)$ is calculated by first transforming the first index of $I_{\mu\nu}^{P}$ to the occupied MO $I_{i\nu}^{P}$ and then performing the update with Γ_{ia}^{P}, again obtained as a matrix multiplication and accumulate on the relative local buffer $L_{b\nu}^{G}(2)$. In this case, the matrices are not sparse, and thus the associated cost is $O(ovn)$ for each P. Finally, the second non-separable contribution to $E_{\mathrm{RI}}^{(2)^x}$ from the 3-center ERI's is computed by integrating the potential $v_H^{\Gamma^P}(\vec{r})$ associated to the $\sum_{\mu\nu} \Gamma_{\mu\nu}^{P} \phi_\mu(\vec{r})\phi_\nu(\vec{r})$ electrostatic density with the auxiliary basis function derivative $\chi_P^x(\vec{r})$

only for the actual P.

At the end of the loop over the auxiliary index P each group stores the two buffers $L^G_{\mu j}(1)$ and $L^G_{b\nu}(2)$ containing the contribution to $L_{\mu j}(1)$ and $L_{b\nu}(2)$ associated to the $P \in [P^{n_P}_{start}, P^{n_P}_{end}]$. In order to obtain $L_{\mu j}(1)$ and $L_{b\nu}(2)$ in their final form, *i.e.* defined over all process, a redistribution step is required, for which each process receives and accumulates from all others the data associated with its new local portion of the two matrices.

Similarly to the case of the computation of B^P_{ia}, the asymptotically dominating steps of this procedure scales as $O(N^4)$. As shown in the pseudocode of figure 8, these are associated to the calculation of Γ^P_{ia} and Γ^{PQ}, update of $L_{b\nu}(2)$ and indices transformations AO$\leftrightarrow$MO. These operations display a relatively small prefactor since they are performed as matrix multiplications. On the other hand the calculation of the integrals and their derivatives is linear scaling for each P since only pairs of overlapping Gaussians need to be considered, and only a finite number of grid points within a spherical region around the center of the primitive Gaussian functions is required. This makes the overall effort in the integral computation $O(N^2)$. Nevertheless this operation displays a quite large prefactor and results in a large amount of the total time (30-40 %) even for relatively large systems.

This part of the algorithm is specific to the computation of the RI-MP2 energy derivatives, meaning that it constitutes an overhead that is not necessary in the case for which only the energy is required. Even if the structure of the described procedure is similar to that employed for the evaluation of B^P_{ia}, in this case not only the integrals are computed, but also their derivatives. This implies a cost for integral computation that is roughly double than that associated to the relative operation in the calculation of B^P_{ia}.

In the case the stress tensor has to be computed, additional operations have to be considered. These operations are not reported in figure 8, but they can be derived by inspection of equations 50 and 51. Note that in this case also elements of the type $(R_{I\beta} - r_\beta)\nabla_{I\alpha}\phi_\mu(\vec{r})\phi_\nu(\vec{r})$ have to be integrated, resulting in an additional overhead roughly equivalent to the computation of the integral derivatives.

4. Final evaluation of $E^{(2)^x}_{\mathbf{RI}}$

The evaluation of the RI-MP2 energy derivatives is completed by a series of operations that allow to generate the $P^{(2)}_{pq}$ and $W^{(2)}_{pq}$ in their final form. Once $P^{(2)}_{pq}$ and $W^{(2)}_{pq}$ are made

Create $P^{(2)}_{pq}$ and $W^{(2)}_{pq}$ parallel distributed matrices defined aver all processes

Fill virtual-virtual block $P^{(2)}_{ab}$, (all a, b)

Fill occupied-occupied block $P^{(2)}_{ij}$ (diagonal elements and $ij \in \{ij\}_{\text{sing}}$)

Transform first index of $L_{\mu i}(1)$ to occupied MO: $L_{ij}(1) = \sum_\mu C_{\mu j} L_{\mu i}(1)$

Complete occupied-occupied block: $P^{(2)}_{ij} = \dfrac{1}{2}\dfrac{L_{ij}(1) - L_{ji}(1)}{\epsilon_j - \epsilon_i}$, $(ij \notin \{ij\}_{\text{sing}})$

Calculate contributions to $W^{(2)}_{pq}$ from $P^{(2)}_{ij}$, $P^{(2)}_{ab}$, $L_{\mu i}(1)$ and $L_{a\nu}(2)$:

$$W^{(2)}_{ij} = \tfrac{1}{2}\left[\sum_\mu C_{\mu j} L_{\mu i}(1) + (\epsilon_i + \epsilon_j) P^{(2)}_{ij}\right]$$
$$W^{(2)}_{ab} = \tfrac{1}{2}\left[-\sum_\nu C_{\nu b} L_{a\nu}(2) + (\epsilon_a + \epsilon_b) P^{(2)}_{ab}\right]$$
$$W^{(2)}_{ai} = -\tfrac{1}{2}\sum_\nu C_{\nu i} L_{a\nu}(2)$$

Calculate Lagrangian L_{bj}: $L_{bj} = \sum_\mu C_{\mu b} L_{\mu j}(1) + \sum_\nu L_{b\nu}(2) C_{\nu j} + \sum_{ac} P^{(2)}_{ac} A_{acbj} + \sum_{ik} P^{(2)}_{ik} A_{ikbj}$

Calculate $P^{(2)}_{ai}$ by solving the Z-vector equations: $\sum_{ai}\left[\delta_{ij}\delta_{ab}(\epsilon_a - \epsilon_i) + A_{aibj}\right] P^{(2)}_{ai} = -L_{bj}$

Complete $P^{(2)}_{pq}$ with occupied-virtual block $P^{(2)}_{ai}$

Complete $W^{(2)}_{pq}$:

$$W^{(2)}_{ij} = W^{(2)}_{ij} + \tfrac{1}{2}\sum_{pq} P^{(2)}_{pq} A_{pqij}$$
$$W^{(2)}_{ai} = W^{(2)}_{ai} + \epsilon_i P^{(2)}_{ai}$$

Transform $P^{(2)}_{pq} \to P^{(2)}_{\mu\nu}$, $W^{(2)}_{pq} \to W^{(2)}_{\mu\nu}$ and symmetrize

Final ${E^{(2)}_{\text{RI}}}^x$ evaluation by contraction of $P^{(2)}_{\mu\nu}$ and $W^{(2)}_{\mu\nu}$ with $F^x_{\mu\nu}$ and $S^x_{\mu\nu}$

Figure 9. Pseudocode of the parallel algorithm for computing $P^{(2)}_{pq}$, $W^{(2)}_{pq}$ and the final contributions to ${E^{(2)}_{\text{RI}}}^x$.

available, their contraction with the skeleton derivatives of the Fock and Overlap matrix elements is performed at the same time with the evaluation of the derivatives of the HF energy leading to the final result.

The sequence of these operations is summarized in the pseudocode of figure 9. The virtual-virtual block of $P^{(2)}_{pq}$ is already available from the procedure described in section A 2, while the occupied-occupied part has to be completed for the non-singular elements according to equation 34. The occupied-virtual block is instead obtained as the solution of the Z-vector equations 27.

In order to do so, first the RI-MP2 specific Lagrangian L_{bj} has to be assembled. As shown in equation 29, four terms contribute to L_{bj}. The first two are calculated from $L_{\mu j}(1)$ and $L_{b\nu}(2)$ just by transforming the indices from the AO to the MO basis. The remaining two are computed by contraction of the the virt-virt and occ-occ blocks of $P^{(2)}_{pq}$ with the integrals generated by coupled-perturbed Hartree-Fock (CPHF) theory (the A_{pqrs} matrix). This contraction is often referred as CPHF-like update, and has a computational cost that is

similar to the update of the Fock matrix in the standard SCF procedure. Equations 36 and 37 display the operations that have to be performed for each CPHF-like update, note that in the actual case the summation ranges are over virt-virt and occ-occ orbitals respectively for the contraction with $P^{(2)}_{ab}$ and $P^{(2)}_{ij}$.

Once L_{bj} is assembled the Z-vector equations are solved employing the Pople method.[63] From a computational standpoint this is equivalent to solving a large system of linear equations with an iterative technique, for which, at each iteration, only the matrix-vector product (CPHF-like update) has to be performed. The parallelization of the CPHF-like update closely follows the scheme employed in CP2K for evaluation of the Fock matrix elements[84,90] and will not be described further here.

Note that, for dense systems with large basis, the computation of the 4-index ERI's over AO, necessary to calculate the exchange part of the CPHF-like update, is by far the most demanding task of this procedure. Since these integrals are the same as those employed in the SCF procedure, if enough memory is available, they can be stored in core, avoiding their recomputation. This can greatly speed up the solution of the Z-vector equations.

The RI-MP2 correction to the energy-weighted density matrix $W^{(2)}_{pq}$ is finally generated from $P^{(2)}_{pq}$, $L_{\mu j}(1)$ and $L_{b\nu}(2)$ according to equations 38-47, for which an additional CPHF-like update is required for the occupied-occupied block. With these matrices defined, previous a step of back-transformation from the AO to the MO basis, the derivatives of the total energy (RI-MP2 + HF) can be finalized by contraction with the skeleton derivatives of the Fock $F^{(x)}_{\mu\nu}$ and Overlap $S^{(x)}_{\mu\nu}$ matrix elements.

5. Memory Usage

The RI-MP2 method, both for energy and derivatives, displays a memory requirement that grows cubically with the system size. This is related to the storage of the B^P_{ia}, Y^P_{ia} and Γ^P_{ia} quantities, while all the other intermediates require at most an $O(N^2)$ memory. An important feature of a parallel algorithm is that, not only the computation, but also the required storage space per task is reduced by increasing the number of processes.

Reported in table IV is the amount of memory that needs to be allocated per MPI task for the storage of the most relevant intermediates. All the cubically demanding quantities are distributed over the total amount of processes (N_p) or, for β^P_{ia} and X^P_{ia}, within the large

Table IV. Memory usage in the different part of the parallel algorithm expressed as "order of" the calculation parameters. n and N_a number of primary and auxiliary basis functions, o and v number of occupied and virtual orbitals, S grid size, N_G and N_w number of groups and group size, N_R and N_r number and size of the replication group ($N_G = N_R N_r$), B_S batch size for ij pairs. N_p number of processes. N_G, N_w and N_p are related by $N_p = N_G N_w$. The notation employed for the entries is referred to the different algorithms.

	Memory
Evaluation of $V_{PQ}^{-1/2}$ and B_{ia}^{P} (A 1):	
$\rho(\vec{r})$, $v(\vec{r})$, $\rho(\vec{G})$, $v(\vec{G})$	S/N_w
$V_{QP}^{-1/2}$ - B_{ia}^{P}	N_a^2/N_p - ovN_a/N_p
$O(N^5)$ Scaling Intermediates (A 2):	
β_{ia}^{P}, X_{ia}^{P} - Y_{ia}^{P}	$ovN_a/(N_r N_w)$ - ovN_a/N_p
A_{aP}^{i}, E_{aP}^{j}, Ξ_{aP}^{i}, Λ_{aP}^{j}	vN_aB_S/N_w
I_{ab}, t_{ab}, $P_{ab}^{(2)}$ - $P_{ij}^{(2)}$	v^2/N_w - o^2
Non-Separable $E_{\mathrm{RI}}^{(2)^x}$, $L_{\mu j}(1)$ and $L_{b\nu}(2)$ (A 3)	
$\rho(\vec{r})$, $v(\vec{r})$, $\rho(\vec{G})$, $v(\vec{G})$	S/N_w
Γ^{PQ} - Γ_{ia}^{P}	N_a^2/N_p - ovN_a/N_p
$\Gamma_{\mu\nu}^{P}$, $I_{\mu\nu}^{P}$ - $\Gamma_{i\nu}^{P}$, $I_{i\nu}^{P}$, $L_{\mu j}^{G}(1)$ - $L_{b\nu}^{G}(2)$	n/N_w - on/N_w - vn/N_w
$L_{\mu j}(1)$ - $L_{b\nu}(2)$	on/N_p - vn/N_p
Final evaluation of $E_{\mathrm{RI}}^{(2)^x}$ (A 4)	
$P_{pq}^{(2)}$, $W_{pq}^{(2)}$ - $P_{ia}^{(2)}$, L_{jb}	n^2/N_p - ov/N_p
CPHF-like update	$N_{\text{AO-ERI}}/N_p$

number of members of each replication group. The algorithm is designed such that the computation, at the process level, involves only the allocation of quadratic intermediates. Nevertheless, these $O(N^2)$ quantities can still require a relatively large amount of memory, for example, the generation of the $(ia|jb)_{\mathrm{RI}}$ integrals implies the product of $v \times N_a$ matrices that, even for medium size systems, would need hundreds of Mb. That's the reason why the group has been introduced, in order to share these objects over more processes. In fact, as shown in table IV, all the quadratically demanding quantities (except for the small $P_{ij}^{(2)}$ matrix) require an amount of memory that is reduced increasing the group size N_w. Moreover, since the actual implementation is based on a hybrid OpenMP/MPI scheme, a similar gain can be achieved by increasing the number of threads employed per MPI task. This leads to more

memory per MPI task without significantly loss of computational efficiency. Which of the two strategies is to be preferred is not obvious since it depends on many aspects such as the machine architecture and the implementation of the parallel libraries. As a rule of thumb, using more processes per group in general leads to better workload distribution within the group, while more threads per MPI task gives better memory management.

The computation of the exchange contribution for each CPHF-like update require the contraction with 4-index ERI's over atomic orbitals $(\mu\nu|\lambda\sigma)$. These are usually calculated only at the first cycle of the procedure and then reused for the subsequent steps. For dense systems employing large basis sets, that is, situations for which the integral screening is not very effective, the storage of these integrals can exceed the amount of memory available per process. In these cases only the largest possible number of ERI's are stored, while the remaining part is computed on the fly at each iteration.

REFERENCES

[1]C. Møller and M. S. Plesset, Phys. Rev. **46**, 618 (1934).

[2]A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry* (McGraw Hill, New York, 1982).

[3]S. Hirata, X. He, M. R. Hermes, and S. Y. Willow, J. Phys. Chem. A **118**, 655 (2014).

[4]S. Grimme, J. Chem. Phys. **124**, 034108 (2006).

[5]L. Goerigk and S. Grimme, J. Chem. Theory Comput. **7**, 291 (2011).

[6]A. Halkier, T. Helgaker, P. Jørgensen, W. Klopper, H. Koch, J. Olsen, and A. K. Wilson, Chem. Phys. Lett. **286**, 243 (1998).

[7]J. J. Shepherd, A. Grüneis, G. H. Booth, G. Kresse, and A. Alavi, Phys. Rev. B **86**, 035111 (2012).

[8]D. Cremer, WIREs Comput. Mol. Sci. **1**, 509 (2011).

[9]J. Almlöf, Chem. Phys. Lett. **181**, 319 (1991).

[10]M. Häser and J. Almlöf, J. Chem. Phys. **96**, 489 (1992).

[11]M. Häser, Theor. Chim. Acta **87**, 147 (1993).

[12]P. Y. Ayala and G. E. Scuseria, J. Chem. Phys. **110**, 3660 (1999).

[13]D. S. Lambrecht, B. Doser, and C. Ochsenfeld, J. Chem. Phys. **123**, 184102 (2005).

[14]B. Doser, D. S. Lambrecht, J. Kussmann, and C. Ochsenfeld, J. Chem. Phys. **130**, 064107 (2009).

[15]P. Y. Ayala, K. N. Kudin, and G. E. Scuseria, J. Chem. Phys. **115**, 9698 (2001).

[16]Y. Jung, Y. Shao, and M. Head-Gordon, J. Comput. Chem. **28**, 1953 (2007).

[17]S. Saebø and P. Pulay, Annu. Rev. Phys. Chem. **44**, 213 (1993).

[18]P. Pulay and S. Saebø, Theor. Chim. Acta **69**, 357 (1986).

[19]G. Rauhut, P. Pulay, and H.-J. Werner, J. Comput. Chem. **19**, 1241 (1998).

[20]M. Schütz, G. Hetzer, and H.-J. Werner, J. Chem. Phys. **111**, 5691 (1999).

[21]G. Hetzer, M. Schütz, H. Stoll, and H.-J. Werner, J. Chem. Phys. **113**, 9443 (2000).

[22]S. Saebø and P. Pulay, J. Chem. Phys. **115**, 3975 (2001).

[23]C. Pisani, M. Busso, G. Capecchi, S. Casassa, R. Dovesi, L. Maschio, C. Zicovich-Wilson, and M. Schütz, J. Chem. Phys. **122**, 094113 (2005).

[24]C. Pisani, L. Maschio, S. Casassa, M. Halo, M. Schütz, and D. Usvyat, J. Comput. Chem. **29**, 2113 (2008).

[25]P. Maslen, Chem. Phys. Lett. **283**, 102 (1998).

[26]P. E. Maslen and M. Head-Gordon, J. Chem. Phys. **109**, 7093 (1998).

[27]S. Y. Willow, K. S. Kim, and S. Hirata, J. Chem. Phys. **137**, 204122 (2012).

[28]S. Y. Willow, M. R. Hermes, K. S. Kim, and S. Hirata, J. Chem. Theory Comput. **9**, 4396 (2013).

[29]D. Neuhauser, E. Rabani, and R. Baer, J. Chem. Theory Comput. **9**, 24 (2013).

[30]Q. Ge, Y. Gao, R. Baer, E. Rabani, and D. Neuhauser, J. Phys. Chem. Lett. **5**, 185 (2014).

[31]W. Klopper, F. R. Manby, S. Ten-No, and E. F. Valeev, Int. Rev. Phys. Chem. **25**, 427 (2006).

[32]C. Hättig, W. Klopper, A. Köhn, and D. P. Tew, Chem. Rev. **112**, 4 (2012).

[33]A. Grüneis, J. J. Shepherd, A. Alavi, D. P. Tew, and G. H. Booth, J. Chem. Phys. **139**, 084112 (2013).

[34]K. Eichkorn, O. Treutler, H. Öhm, M. Häser, and R. Ahlrichs, Chem. Phys. Lett. **240**, 283 (1995).

[35]M. Feyereisen, G. Fitzgerald, and A. Komornicki, Chem. Phys. Lett. **208**, 359 (1993).

[36]H.-J. Werner, F. R. Manby, and P. J. Knowles, J. Chem. Phys. **118**, 8149 (2003).

[37]A. F. Izmaylov and G. E. Scuseria, Phys. Chem. Chem. Phys. **10**, 3421 (2008).

[38]L. Maschio, D. Usvyat, F. R. Manby, S. Casassa, C. Pisani, and M. Schütz, Phys. Rev. B **76**, 075101 (2007).

[39]D. Usvyat, L. Maschio, F. R. Manby, S. Casassa, M. Schütz, and C. Pisani, Phys. Rev. B **76**, 075102 (2007).

[40]M. Katouda and S. Nagase, J. Chem. Phys. **133**, 184103 (2010).

[41]Y. Jung, R. C. Lochan, A. D. Dutoi, and M. Head-Gordon, J. Chem. Phys. **121**, 9793 (2004).

[42]A. M. Burow, M. Sierka, and F. Mohamed, J. Chem. Phys. **131**, 214101 (2009).

[43]F. Weigend, M. Häser, H. Patzelt, and R. Ahlrichs, Chem. Phys. Lett. **294**, 143 (1998).

[44]D. E. Bernholdt and R. J. Harrison, J. Chem. Phys. **109**, 1593 (1998).

[45]F. Weigend, A. Köhn, and C. Hättig, J. Chem. Phys. **116**, 3175 (2002).

[46]A. C. Limaye and S. R. Gadre, J. Chem. Phys. **100**, 1303 (1994).

[47]A. M. Marquez and M. Dupuis, J. Comput. Chem. **16**, 395 (1995).

[48]I. M. B. Nielsen and E. T. Seidl, J. Comput. Chem. **16**, 1301 (1995).

[49]J. Baker and P. Pulay, J. Comput. Chem. **23**, 1150 (2002).

[50]K. Ishimura, P. Pulay, and S. Nagase, J. Comput. Chem. **27**, 407 (2006).

[51]D. E. Bernholdt and R. J. Harrison, Chem. Phys. Lett. **250**, 477 (1996).

[52]M. Katouda and S. Nagase, Int. J. Quantum Chem. **109**, 2121 (2009).

[53]B. Doser, D. S. Lambrecht, and C. Ochsenfeld, Phys. Chem. Chem. Phys. **10**, 3335 (2008).

[54]E. F. Valeev and C. L. Janssen, J. Chem. Phys. **121**, 1214 (2004).

[55]Y. Nakao and K. Hirao, J. Chem. Phys. **120**, 6375 (2004).

[56]I. M. B. Nielsen and C. L. Janssen, J. Chem. Theory Comput. **3**, 71 (2007).

[57]L. Maschio, J. Chem. Theory Comput. **7**, 2818 (2011).

[58]M. Del Ben, J. Hutter, and J. VandeVondele, J. Chem. Theory Comput. **8**, 4177 (2012).

[59]M. Del Ben, J. Hutter, and J. VandeVondele, J. Chem. Theory Comput. **9**, 2654 (2013).

[60]M. D. Ben, O. Schtt, T. Wentz, P. Messmer, J. Hutter, and J. VandeVondele, Computer Physics Communications **187**, 120 (2015).

[61]M. Del Ben, M. Schönherr, J. Hutter, and J. VandeVondele, J. Phys. Chem. Lett. **4**, 3753 (2013).

[62]M. Del Ben, M. Schönherr, J. Hutter, and J. VandeVondele, J. Phys. Chem. Lett. **5**, 3066 (2014).

[63]J. A. Pople, R. Krishnan, H. B. Schlegel, and J. S. Binkley, Int. J. Quantum Chem. Symp. **13**, 225 (1979).

[64]E. Kraka, J. Gauss, and D. Cremer, J. Mol. Struct.: THEOCHEM **234**, 95 (1991).

[65]T. Helgaker, P. Jørgensen, and N. Handy, Theor. Chim. Acta **76**, 227 (1989).

[66]C. M. Aikens, S. P. Webb, R. L. Bell, G. D. Fletcher, M. W. Schmidt, and M. S. Gordon, Theor. Chem. Acc. **110**, 233 (2003).

[67]F. Weigend and M. Häser, Theor. Chem. Acc. **97**, 331 (1997).

[68]M. J. Frisch, M. Head-Gordon, and J. A. Pople, Chem. Phys. Lett. **166**, 275 (1990).

[69]M. Head-Gordon, Mol. Phys. **96**, 673 (1999).

[70]R. A. Distasio, R. P. Steele, Y. M. Rhee, Y. Shao, and M. Head-Gordon, J. Comput. Chem. **28**, 839 (2007).

[71]I. M. Nielsen, Chem. Phys. Lett. **255**, 210 (1996).

[72]G. D. Fletcher, A. P. Rendell, and P. Sherwood, Mol. Phys. **91**, 431 (1997).

[73]K. Ishimura, P. Pulay, and S. Nagase, J. Comput. Chem. **28**, 2034 (2007).

[74]C. Hättig, A. Hellweg, and A. Köhn, Phys. Chem. Chem. Phys. **8**, 1159 (2006).

[75]G. Lippert, J. Hutter, and M. Parrinello, Theor. Chem. Acc. **103**, 124 (1999).

[76]M. Krack and M. Parrinello, Phys. Chem. Chem. Phys. **2**, 2105 (2000).

[77]S. Hirata and S. Iwata, J. Chem. Phys. **109**, 4147 (1998).

[78]The CP2K developers group, CP2K is freely available from: http://www.cp2k.org/ (2014).

[79]M. Head-Gordon, J. A. Pople, and M. J. Frisch, Chem. Phys. Lett. **153**, 503 (1988).

[80]J. L. Whitten, J. Chem. Phys. **58**, 4496 (1973).

[81]B. I. Dunlap, J. W. D. Connolly, and J. R. Sabin, J. Chem. Phys. **71**, 3396 (1979).

[82]O. Vahtras, J. Almlöf, and M. Feyereisen, Chem. Phys. Lett. **213**, 514 (1993).

[83]G. Lippert, J. Hutter, and M. Parrinello, Mol. Phys. **92**, 477 (1997).

[84]J. VandeVondele, M. Krack, F. Mohamed, M. Parrinello, T. Chassaing, and J. Hutter, Comput. Phys. Commun. **167**, 103 (2005).

[85]G. J. Martyna and M. E. Tuckerman, J. Chem. Phys. **110**, 2810 (1999).

[86]P. E. Blöchl, J. Chem. Phys. **103**, 7422 (1995).

[87]L. Genovese, T. Deutsch, A. Neelov, S. Goedecker, and G. Beylkin, J. Chem. Phys. **125**, 074105 (2006).

[88]S. Goedecker, M. Teter, and J. Hutter, Phys. Rev. B **54**, 1703 (1996).

[89]M. Guidon, F. Schiffmann, J. Hutter, and J. VandeVondele, J. Chem. Phys. **128**, 214104 (2008).

[90]M. Guidon, J. Hutter, and J. VandeVondele, J. Chem. Theory Comput. **5**, 3010 (2009).

[91]J. Gerratt and I. M. Mills, J. Chem. Phys. **49**, 1719 (1968).

[92]N. C. Handy and H. F. Schaefer, J. Chem. Phys. **81**, 5031 (1984).

[93]T. J. Lee, S. C. Racine, J. E. Rice, and A. P. Rendell, Mol. Phys. **85**, 561 (1995).

[94]Y. Osamura, Y. Yamaguchi, P. Saxe, D. Fox, M. Vincent, and H. Schaefer, J. Mol. Struct.: THEOCHEM **103**, 183 (1983).

[95]V. Weber and C. Daul, Chem. Phys. Lett. **370**, 99 (2003).

[96]T. Abe, Y. Sekine, and F. Sato, Chem. Phys. Lett. **557**, 176 (2013).

[97]H. Weiss, R. Ahlrichs, and M. Häser, J. Chem. Phys. **99**, 1262 (1993).

[98]O. H. Nielsen and R. M. Martin, Phys. Rev. Lett. **50**, 697 (1983).

[99]O. H. Nielsen and R. M. Martin, Phys. Rev. B **32**, 3780 (1985).

[100]D. Marx and J. Hutter, in *Ab initio molecular dynamics: Theory and implementation* (Modern Methods and Algorithms of Quantum Chemistry, John Von Neumann Institute for Computing, Forschungszentrum Julich, 2000) pp. 329–477.

[101]J. Schmidt, J. VandeVondele, I.-F. W. Kuo, D. Sebastiani, J. I. Siepmann, J. Hutter, and C. J. Mundy, J. Phys. Chem. B **113**, 11959 (2009).

[102]A. Dal Corso and R. Resta, Phys. Rev. B **50**, 4327 (1994).

[103]L. C. Balbás, J. L. Martins, and J. M. Soler, Phys. Rev. B **64**, 165110 (2001).

[104]K. Doll, R. Dovesi, and R. Orlando, Theor. Chem. Acc. **112**, 394 (2004).

[105]T. H. Dunning, J. Chem. Phys. **90**, 1007 (1989).

[106]D. E. Woon and T. H. Dunning, J. Chem. Phys. **98**, 1358 (1993).

[107]J. Paier, C. V. Diaconu, G. E. Scuseria, M. Guidon, J. VandeVondele, and J. Hutter, Phys. Rev. B **80**, 174114 (2009).

[108]J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. **77**, 3865 (1996).

[109]F. H. Allen, Acta Crystallogr., Sect. B: Struct. Sci **58**, 380 (2002).

[110]L. Maschio, D. Usvyat, M. Schütz, and B. Civalleri, J. Chem. Phys. **132**, 134706 (2010).

[111]L. Maschio, B. Civalleri, P. Ugliengo, and A. Gavezzotti, J. Phys. Chem. A **115**, 11179 (2011).

[112]A. W. Hewat and C. Riekel, Acta Crystallogr. Sect. A **35**, 569 (1979).

[113]R. Boese, N. Niederprüm, D. Bläser, A. Maulitz, M. Y. Antipin, and P. R. Mallinson, J. Phys. Chem. B **101**, 5794 (1997).

[114]L. L. Shipman, A. W. Burgess, and H. A. Scheraga, J. Phys. Chem. **80**, 52 (1976).

[115]W. Keesom and J. Kohler, Physica **1**, 655 (1934).

[116]A. Curzon, Physica **59**, 733 (1972).

[117]A. Simon and K. Peters, Acta Crystallogr. Sect. B **36**, 2750 (1980).

[118]C. Müller and D. Usvyat, J. Chem. Theory Comput. **9**, 5590 (2013).

[119]P. J. Bygrave, N. L. Allan, and F. R. Manby, J. Chem. Phys. **137**, 164102 (2012).

[120]O. Sode, M. Keceli, K. Yagi, and S. Hirata, J. Chem. Phys. **138**, 074501 (2013).

[121]S. Wen and G. J. O. Beran, J. Chem. Theory Comput. **7**, 3733 (2011).

[122]S. Grimme, J. Antony, S. Ehrlich, and H. Krieg, J. Chem. Phys. **132**, 154104 (2010).

[123]P. Jurecka, J. Sponer, J. Cerny, and P. Hobza, Phys. Chem. Chem. Phys. **8**, 1985 (2006).

[124]T. Takatani, E. G. Hohenstein, M. Malagoli, M. S. Marshall, and C. D. Sherrill, J. Chem. Phys. **132**, 144104 (2010).

[125]S. Tosoni, C. Tuma, J. Sauer, B. Civalleri, and P. Ugliengo, J. Chem. Phys. **127**, 154102 (2007).

[126]S. Hirata, J. Chem. Phys. **129**, 204104 (2008).

[127]K. E. Riley and P. Hobza, J. Phys. Chem. A **111**, 8257 (2007).

[128]M. Del Ben, J. VandeVondele, and B. Slater, J. Phys. Chem. Lett. **5**, 4122 (2014).

[129]F. Neese, T. Schwabe, and S. Grimme, J. Chem. Phys. **126**, 124115 (2007).

[130]R. C. Lochan, Y. Shao, and M. Head-Gordon, J. Chem. Theory Comput. **3**, 988 (2007).

[131]H. Ji, Y. Shao, W. A. Goddard, and Y. Jung, J. Chem. Theory Comput. **9**, 1971 (2013).

[132]A. M. Burow, J. E. Bates, F. Furche, and H. Eshuis, J. Chem. Theory Comput. **10**, 180 (2014).